



Automating Patch Management

Protecting Your Network and
Your Bottom Line

SUMMARY

The Aberdeen Group reports that business and government entities spend more than \$2 billion per year to identify, research and deploy security patches to enterprise devices. Authentium PatchMatrix is the market-leading technology, and the only approach that enables both agent, and agentless, patch management.

Authentium, Inc. is one of the fastest-growing companies in the security software industry, servicing more than 2,200 enterprise, government, and education customers and 2,200,000 end users licenses worldwide.

DOCUMENT INFORMATION

Author	: Solutions Marketing
Version	: 03.03.04_(printed 10 Mar. 04 18:43)
Classification	: Public

INTERNATIONAL COPYRIGHT AND CONFIDENTIALITY NOTICE

Copyright © 2004 Authentium, Inc. All rights reserved.

This document and the information contained herein is the subject of copyright and intellectual property rights under international convention. All rights reserved. . Information in this document is subject to change without notice. This document may be distributed freely only in whole, however no alterations are allowed without the expressed written consent of the author, Authentium, Inc.

This document is for informational purposes only AUTHENTIUM, INC. MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

PatchMatrix is a registered trademark of Authentium. Authentium and the Authentium logo are trademarks of Authentium, Inc. Microsoft, Windows, Windows NT, Windows 2000, Windows 2003, .NET, SQL Server and Windows XP are either registered trademarks or trademarks of Microsoft Corporation in the USA and other countries. All other product and corporate names may be trademarks or registered trademarks, and are used only for identification, without intent to infringe.

Note: This document contains descriptions of networks and their configurations. Some obvious configuration information is omitted to make this document easy to read. The reader is responsible for implementation.

For more information about Authentium and PatchMatrix, check us out on the Web at <http://www.authentium.com> and <http://www.gopm.com>.

*Authentium, Inc
7121 Fairway Drive
Palm Beach Gardens FL 33418*

*Call 1-800-423-9167
Fax 1-561-575-3026*

www.authentium.com

CONTENTS

CONTENTS.....	1
ABOUT AUTHENTIUM.....	2
CORPORATE BACKGROUND	2
EXECUTIVE SUMMARY	3
IMPACT ON COMPUTER USERS	3
THE CHALLENGE.....	4
NETWORK VULNERABILITY	4
The Growing Number of Patches and Service Packs	4
The Problem with Patches	5
The Juggling Act of Patch Management	5
The Dangers of a Vulnerable Network	6
Why Patch Management Has Historically Not Been Adequately Addressed	7
SOLVING THE PATCH MANAGEMENT DILEMMA.....	9
COMPARISON OF APPROACHES	9
Manual Solutions – The “Sneakernet” Approach	9
Home-Grown Remote Solutions	9
Microsoft Solutions	10
Third Party Solutions	11
MAKING THE DECISION	11
Agent Versus Agentless Operation	11
Supporting Disconnected Networks	12
The Patch Management Database Issue	12
OVERVIEW OF AUTHENTIUM PATCHMATRIX.....	14
AUTOMATING THE PATCH MANAGEMENT PROCESS	14
OVERVIEW OF HOW PATCHMATRIX WORKS	14
BENEFITS OF PATCHMATRIX	15
Ease of Use	15
Flexibility – Agent or Agentless Operation, or Both	16
Superior Validation Process	17
Comprehensive Patch Management Database	18
Independent Validation of Patch Interdependencies	19
SUMMARY OF FEATURES	20
PATCHMATRIX FEATURES AND VALUE	20
CONCLUSIONS	21
APPENDIX A:	23
PATCHMATRIX COMPONENTS	23
EXAMPLE OF PATCHMATRIX OPTIONAL AGENT ARCHITECTURE	24

ABOUT AUTHENTIUM



CORPORATE BACKGROUND

Authentium, Inc. is one of the fastest-growing companies in the security software industry, servicing more than 2,200 enterprise, government, and education customers and 2,200,000 end users licenses worldwide. The company is privately held. The company's institutional investors include Mindport and VeriSign.

Authentium's customer base includes leading government agencies, such as the Australian Bureau of Statistics, US Bureau of Census, Dept of Agriculture, Dept. of Commerce, Dept. of Prisons, Dept of Treasury, the FCC (Federal Communications Commission), and the Nuclear Energy Institute.

Customers of Authentium AV Matrix (Command Antivirus), PatchMatrix Patch Management, Galileo Central Management, and Jupiter security appliance products include EMC, Microsoft, Nielsen, Nikon, Raytheon, Sears, Sungard, and Unisys. Education customers include more than 1,000 schools and major universities, such as Oklahoma State, Tufts University, UCLA, the University of Pittsburg, the University of Texas, and New York State Public Schools.

Customers for the new Authentium ESP (Enhanced Security Platform) range of plug-in security applications (including antivirus/firewall/spam filtering/identity fraud management/parental controls) include the world's largest broadband ISP, Comcast/AT&T, with 4.5 million users, and Riverdeep, the largest supplier of educational software, with more than 45,000 schools as clients.

CORPORATE STRUCTURE

Authentium is incorporated in Delaware, with wholly-owned subsidiaries based in Palm Beach Gardens, Florida, London, England and Melbourne, Australia. All intellectual property is US-based.

For the purposes of US privacy regulation and customer information, all customer data managed by Authentium authentication technology is hosted, maintained, and owned by the customer.

All publicly-accessible data is available from the company's web site at www.authentium.com.

EXECUTIVE SUMMARY

IMPACT ON COMPUTER USERS

The Aberdeen Group reports that business and government entities spend more than **\$2 billion per year** to identify, research and deploy security patches to enterprise devices (Aberdeen Group White Paper, November 2002). Yet, patch management has traditionally been a low visibility, labor-intensive duty, with few rewards.

Frequently a manual operation that involves days to perform, the process all too often takes a back seat to more obvious priorities. Despite the expenditures that companies invest in the activity, traditional patch management practices are no longer effective against the overwhelming incidence of virus attacks and other vulnerabilities now infecting the enterprise.

Research analyst firm Gartner questioned the reliability of a manual approach to patch management in a recent Research Note entitled "Patch Management: Identifying the Vendor Landscape." The March 18, 2003 note states the following:

"There is nothing new about the need to manage the software that is deployed on servers and PCs. There is, however, a new appreciation of the risks associated with not applying the growing number of software patches.

Enterprises have renewed their focus on the long-standing problem of patch management in an effort to reduce the exposure to mass outages or security breaches. Patch management is an area in which manual approaches have no chance of being effective.

The bottom line is that companies who cling to the old ways of doing patch management are at risk of endangering their mission-critical systems and databases, and therefore the viability of the company itself."

This paper provides Information Technology executives and administrators with an overview of the need for effective patch management as well as the strengths and weaknesses of available automated patch management solutions. It also reviews the specific benefits of Authentium's automated patch management tool: PatchMatrix™.

THE CHALLENGE

NETWORK VULNERABILITY

System administrators face the daunting challenge of keeping systems up to date and ensuring that operating systems and applications are current. This includes deploying patches to systems that are vulnerable. These patches fall into two broad categories: hot-fixes and service packs.

The number and complexity of patches present a challenge for administrators. How do patches work together? Can they be deployed on top of one another? Endless questions abound, while the administrator is faced with the prospect of testing, programming and praying that patch deployments succeed. Meanwhile, networked systems are vulnerable to external and internal security threats. Systems need to be patched, but patch management is a full-time job requiring specialized knowledge.

The Growing Number of Patches and Service Packs

A software patch repairs a known problem (bug) in a program, whether in a Windows operating system or an application. Some of these bugs cause little or no concern (i.e., software does not “hang” and systems are not vulnerable as a result). Other bugs cause catastrophic failures or give a malevolent person the ability to gain access to a system and even be granted administrative privileges.

A **patch** is an immediate solution provided to users once a bug is discovered; it can often be downloaded from the software maker's Web site. The problem that it repairs may relate to serious security vulnerability or system instability or simply to a cosmetic change, such as a misspelled word within a prompt. In the case of a serious performance issue, the patch may replace core system files, with possible interactions or conflicts with other files. It may also alter the system's Registry, with other potential ramifications.

Hot-fixes are released frequently, without any advance notice. Hot-fixes are typically not fully regression tested.

Service packs are released much less frequently than hot-fixes, and include improvements and enhancements to operation or performance of a program, in addition to including most hot-fixes issued since the last service pack or product version. These changes have been regression tested to ensure a minimal level of compatibility. For the OS, service packs are essentially new OS deployments and require much planning. On the other hand, hot-fixes are typically unplanned deployments. It is not safe to assume that a company can ignore hot-fixes, and just wait for service packs to arrive. The reason is that many hot-fixes are released to

repair a newly discovered security hole. If ignored, serious consequences may result.

The Problem with Patches

One problem with patches is keeping abreast of what is out there. To keep current, administrators must periodically check for updates at the Microsoft support Web site. This, by itself, is a time consuming activity. From the Web site, an administrator is supposedly able to determine whether a particular hot-fix or service pack is a candidate for implementation on his/her network. Unfortunately, that supposition is seldom the case.

The reason is that there are too many patches of which to keep track. With thousands of software patches for NT/2000/XP and many business applications released by Microsoft, it is impossible for IT and security professionals to be knowledgeable enough to ensure quality deployments.

More importantly, testing is not always thorough enough before patches are released. As an example, in February 2002, Microsoft released a security patch for Internet Explorer. After only a few hours on the support Web site, they withdrew the fix because of an error in it. Several days later, they released it again. This drew much criticism from the public, with pressure on Microsoft to test its fixes more stringently.

The sheer number of updates – coupled with the fact that patches can wreak havoc with a system if not properly tested – leaves many system managers unsure about what to do.

Yet another problem is that not all machines will require all patches. Administrators of multiple server enterprises find themselves in the situation of keeping track of different levels of patches for different configurations of machines under their supervision.

The Juggling Act of Patch Management

Administrators must complete the patch management task – to include researching what patches are available, testing the patch, figuring out which machines need it, then deploying the patch and validating that it was done correctly – while still maintaining quality of service (QoS) in other areas. Hot-fixes are not announced in advance when released, so the administrator cannot plan for the activity. This

forces the administrator to drop everything in order to attend to the urgent security issue, a situation that can clearly have a negative impact on QoS.

In addition, there are risks associated with updating systems (just as there are risks with not updating the system). This risk is increased when an urgent update, such as a security hot-fix, is required. Added to the problem of maintaining QoS is the possibility that the system may become unstable once a hot-fix is installed. These risks include complete system lockup and other failures. For this reason, even in the face of an urgent update, an administrator will want to thoroughly test the fix before deploying it.

Time management becomes an obvious component of the patch management challenge. The process of testing and patching numerous systems over an entire enterprise is extremely time consuming, especially if the administrator cannot automate the deployment process.

In the end, the patch management challenge becomes a fine balancing act. Administrators must weigh the consequences of each possible course of action: to live with the risks of a vulnerable network or intelligently test and deploy patches and live with the reduction in QoS and the potential risk of a patch-related catastrophe.

The Dangers of a Vulnerable Network

Despite prevailing practices and conventional wisdom, the security and stability of today's corporate computer systems depend more on fixing security holes in an enterprise's operating system and applications and less on deploying antivirus, firewalls, intrusion detection devices and other expensive, cutting-edge security technology.

As evidence, in a February 10, 2003 article entitled "Feds Look for Lessons From Internet Worm," Federal Computer Week reports that 98 percent of successful hacker attacks are the result of known software bugs or other vulnerabilities that could have been fixed with available patches. Yet, only 2 percent of the companies surveyed had deployed these patches.

Awareness about security vulnerabilities has heightened in the aftermath of the widespread damage caused by the "SQL Slammer" worm and the earlier "Nimda." SQL Slammer, also called "Sapphire," exploited servers using unpatched Microsoft SQL Server as well as workstations running Microsoft Desktop Engine (MDE) 2002.

Nimda attacked unpatched servers running Internet Information Server (IIS) and desktops using the IIS personal Web server. Both appear to have taken advantage of known vulnerabilities that had been exposed and widely publicized months earlier. In both situations, hot-fixes were readily available that could have warded off problems. These events, and others, have triggered a cultural change in the IT departments of corporate America. As a result, many IT professionals have taken a more proactive stance by searching for ways to deploy hot-fixes and plug security holes before disaster strikes.

Many system administrators want to patch proactively, but they do not have confidence that the right solution exists – one that reliably navigates them through the complex maze that is today's patch management dilemma.

Why Patch Management Has Historically Not Been Adequately Addressed

The majority of businesses traditionally failed to address the vital security requirement of patch management for several reasons:

- **Risk to uptime.** First, there is the innate fear by system administrators that deploying patches can introduce instability. These administrators have incentive to keep their systems running (uptime). Often, one sees a business proudly proclaiming that they have “99.9% uptime.” As a result, it is easily perceived that unlimited uptime produces better business results. Security holes (the result of software flaws) bring a business down, but the patch process can also cause havoc. Thus, we see a conflict between improving security practices and keeping a system up 24x7.
- **Lack of available resources.** Software bugs are often only found in specific configurations, making fixes difficult to predict and deploy. Tracking down all the servers and workstations in a large enterprise and checking them for the proper patch levels and configurations requires an enormous amount of legwork, but that's only the beginning. Few companies can afford to devote the full-time efforts of their personnel to the testing of various software patch combinations in order to deploy patches reliably over time. The interdependencies of patches are so complex that it is not surprising to see cavalier attitudes from administrators (like, “If it isn't broken, don't fix it”). In other words, administrators ask themselves, “Why take the risk of patching if that action leads to downtime?” This objection comes down to a simple reality that complexity adversely affects quality.

-
- **Patch management requires specialist knowledge.** In addition, managing the volume of patches available is difficult and requires specialized knowledge. Many questions arise.
 - There are so many patches. What patches should I deploy?
 - What happens if the patches are overwritten or rendered useless by some future software installations?
 - If patches conflict somehow, can I avoid these pitfalls?

SOLVING THE PATCH MANAGEMENT DILEMMA

COMPARISON OF APPROACHES

Traditional approaches to solving the patch management dilemma fall into three categories: manual in-house solutions, homegrown solutions and external solutions. The latter includes reliance on Microsoft's processes to keep track of and deploy patches as well as the implementation of patch management software from third-party vendors specializing in solutions that protect Windows-based platforms from system intruders.

Manual Solutions – The “Sneakernet” Approach

The majority of businesses Conventional methods of assessment and remediation often involve personnel manually researching and patching the holes in each server and workstation separately. Even in a small enterprise, this approach can take days of effort.

A good portion of this time is spent updating systems by walking from computer to computer (“Sneakernet”). This laborious method compounds the time needed to research patches, to keep up to date on the latest array of patches and to perform the extensive testing required to monitor the patch process on a small subset of machines before deploying the patch to the entire organization.

The bottom line: The process of manually researching and deploying the patches to every server and workstation separately is simply too time consuming and error prone to be a viable option for most companies.

Home-Grown Remote Solutions

Some companies opt to develop their own automated processes to check for and fix software bugs. This approach requires extensive knowledge and programming skills. It relies on scripting a deployment engine or framework product with custom code, which can take valuable hours to develop and test – time that IT staffers could spend on other critical issues.

Both manual remediation and homegrown solutions are costly, unreliable and inefficient ways to repair security holes. In order for patch deployment to be effective, it needs to be a scheduled event that can deploy patches reliably and with a validation process at the end that ensures that the update has gone as planned and continues to be installed properly.

Microsoft Solutions

Microsoft provides three automated solutions to patch management and deployment: HFNetChk, Microsoft SMS and Microsoft Software Update Services.

- **HFNETCHK** is provided free of charge for Microsoft customers from both the Shavlik and Microsoft Support Web sites. HFNetChk will inventory a computer or group of computers remotely from a command line for the absence or presence of security patches. HFNetChk covers Windows NT4/2000/XP/Server 2003 as well as IIS 4.0, IIS 5.0, SQL Server 7.0 and SQL Server 2000 (including MS Data Engine), and Internet Explorer 5.01 or later. HFNetChk checks each system for patches based on the registry key that is associated with the patch as well as the file version and file checksum for each file in the patch. If everything matches, the patch is considered installed. In order to automate the actual deployment of a patch, customers must purchase the Shavlik tool HFNetChkPRO.
- **MICROSOFT SMS FEATURE PACK** can be used to distribute patches automatically. SMS is designed for use in medium- to large-sized organizations. It is expensive and requires an SMS installer on every client system. SMS is more of an asset manager, though it does have some deployment capability. It requires extensive knowledge and does not have a comprehensive set of patches when compared to other solutions, such as Authentium's PatchMatrix™, discussed later in this paper. SMS takes a policy-oriented approach to the deployment of patches, giving the enterprise more flexibility than the SUS tool discussed next.
- **MICROSOFT SOFTWARE UPDATE SERVICES (SUS)** is designed for small- to medium-sized organizations that have not yet deployed Microsoft Systems Management Server (SMS). This is a free product using a server-based model with client agents on all machines, which automate the process of deployment. It runs on a localized server and synchronizes with the Windows support Web site (Update site) to deliver all of the updates to all Windows 2000 and XP licensed machines. SUS enables administrators to deploy these updates. Using workgroup policies, SUS approaches patch management with a global enforcement philosophy rather than a policy-oriented approach (like SMS does).

Third Party Solutions

For mid to large-sized companies, automated tools from third parties, such as Authentium, can reduce both risks and costs associated with patch management, and increase productivity, while reducing liabilities and workload.

MAKING THE DECISION

According to Gartner Research (April 24, 2002 research report), the following questions should be asked when deciding whether to adopt a third-party solution:

- What is the security risk involved if patches are not applied?
- What will it cost to stay abreast of the latest patches and download them to each system?
- Would it be less expensive to deploy hot-fixes from a central server rather than manually installing them on each machine?
- Can manual installation be easily validated to determine that the patches have been installed and are properly functioning?

When evaluating which third-party patch management tool to employ, companies should address broad issues relating to flexibility and reliability as well as feature sets. There are two broad issues to keep in mind when assessing alternatives:

1. whether to use an agent versus a agentless based solution (or one that offers both); and
2. the scope and reliability of the patch management database that supports the product.

Agent Versus Agentless Operation

Entirely agentless solutions, such as Gravity Storm, Shavlik and Ecora, require domain or local administrator access along with other requirements that limit the types of networks in which they can be used. They are most effectively used in systems where the administrator has a high degree of control over target systems such as the server room. Unfortunately for non-agent solutions, the most important systems are the ones that are usually the hardest to reach (locked down or

otherwise isolated).

Entirely agent-based solutions deploy client agents on each target workstation/server (PatchLink Update, BigFix Enterprise). The agent periodically polls the patch server for new updates, so roaming user issues can be addressed. Agent-based products are more difficult to set up and integrate, and are therefore more costly, but they provide more functionality and offer greater control over network bandwidth usage.

Authentium's PatchMatrix is the only solution that offers both agent and agentless modes of operations. This capability supports customers who have the requirement to manage servers and workstations that are isolated or otherwise locked down, which require the use of an agent. In these enterprises, the non-agent solution can be deployed for most workstations, with the optional client agent deployed on the locked-down machines. (See later discussion in this paper under "Overview of the Main Benefits of Authentium PatchMatrix.")

Supporting Disconnected Networks

Since patch management solutions require regular updates for critical patch data, it is impractical to support systems from a network with no connections to the outside world. PatchMatrix offers an exclusive feature that allows administrators to replace being connected to the outside world with portable "packaged updates" functionality. This means that the administrator can manage machines orphaned from the Internet by using portable media and a simple process.

The Patch Management Database Issue

Most patching tools use a limited set of patches and patch information supplied from public sources (such as MSSecure.XML). Free patch detection tools such as HFNetChk available from the Microsoft Web site, and most third-party patch management solutions, use a common database of patches. This database is supplied to the public and is limited in scope. Not all critical patches are included, nor are all applications supported. Another shortfall is that public databases do not include the ability to add proprietary patches. In addition, testing is limited and does not include analysis of patch supercedence, prerequisites, co-requisites or conflict issues related to coexisting patches shared by more than one application.

The patch management databases of third-party vendors fall into two groups:
solution:

- Those that use a public source database and test just one patch at a time.
- Those that have a third-party patch database and do comprehensive, independent testing on a combination of patches and deployment scenarios.

When assessing patch management tools, companies will want to know what type of patch database (public or private) underlies the tool and how patches are tested, one at a time or in combination.

OVERVIEW OF AUTHENTIUM PATCHMATRIX

AUTOMATING THE PATCH MANAGEMENT PROCESS

Authentium PatchMatrix is a software patch management tool for networked workstations and servers. PatchMatrix discovers patches that are missing and applicable and reliably deploys them. Administrators define a baseline of required patches for all systems and report the conformance of managed systems against this baseline.

PatchMatrix features an exclusive, meta-database of patch information and instructions that are maintained by Authentium's software patch experts. By helping an enterprise identify missing patches and remediate with confidence, PatchMatrix enforces software security policies and provides a superior way of managing hot-fixes, patches and service packs.

Supporting Windows NT/2000/XP, SQL Server, IE and a long list of mission-critical applications, PatchMatrix enables a company to research available fixes, scan workstations and servers, deploy updates to any number of networked machines without remote agents, and validate that the job was done right. PatchMatrix also supports an optional client agent to enable companies to deploy both an agent-based and agentless method of patch management.

OVERVIEW OF HOW PATCHMATRIX WORKS

From one management console, PatchMatrix simplifies the remediation inventory process; researches the complexities, possible interdependencies, and possible conflicts of potential updates; and deploys hot-fixes and service packs throughout the entire enterprise without requiring an agent.

The steps to this process are described below:

- The system administrator chooses which machines to manage.
- PatchMatrix uses Remote Procedure Calls (RPC) to manage machines for systems not using client agents. It also allows the system administrator to install an optional agent (using encrypted TCP/IP) when it is desirable, on selected machines. The agent feature is discussed later in this paper in more detail.
- PatchMatrix discovers patch levels for all managed machines' operating systems and applications.

-
- With a user-defined list of required patches, PatchMatrix compares software patches against this baseline of required patches. The result is a conformance report for all managed machines.
 - Using the vast patch management knowledge base that is maintained by Authentium's team of experts in support of the product, it identifies which updates are applicable and, most important, which ones can be safely implemented given the configuration of the total system and the repairs being considered.
 - It makes recommendations on how to reliably update the machines with patches and warns/prevents impending doom with mismatches, missing prerequisites, etc.
 - PatchMatrix then automates the patch management process of managing and deploying updates over the network to individual workstations and servers. Remediation can be customized, for example, accomplished by department or by individual workstation or server or configured to begin after hours.
 - Finally, PatchMatrix validates the process and delivers executive and technical Conformance Reports based on a user-defined set of required patches (baseline), summarizing all actions taken, during and after remediation

For a more technical discussion about PatchMatrix's components, including the Master Agent and Console, and possible configurations, please refer to Appendix A, p.16.

BENEFITS OF PATCHMATRIX

The main benefits of Authentium PatchMatrix are ease of use, ease of installation, and Total Cost of Ownership.

Ease of Use

PatchMatrix allows IT managers to easily install patches remotely on networked machines. It is easy to use for two reasons:

- The meta-database includes instructions and patch information so that the

guesswork is eliminated from the patch management process.

- The architecture is designed to scale to any size business.

The meta-database prevents mistakes from happening during the deployment phase of patching. While patches themselves must be tested against legacy software, the deployment can be extremely complex. The number of patches and their interdependencies make it impossible to keep up and ensure a reliable patch management practice. PatchMatrix is specialized to support these needs with a reliable, automatic meta-database.

The components of Authentium PatchMatrix are designed to be installed minimally and only where needed. A Master Agent can be installed on a server or on a workstation for smaller installations. The management console can be installed as many times as desired. The management consoles are connected to the Master Agent. Finally, the optional client agent (“Leaf Agent”) is available for companies who do not want to use RPC. Large or small, the PatchMatrix architecture is flexible and easy to maintain for any organization.

Flexibility – Agent or Agentless Operation, or Both

PatchMatrix includes a new Optional Agent to provide system administrators with an enhanced level of flexibility to manage patches in the most optimal manner for their organization’s unique computing environment. As previously discussed, PatchMatrix does not require client agents on target machines. Companies that do not want agent software installed on individual workstations can implement PatchMatrix in the standard non-agent deployment using a variety of configurations. The non-agent solution minimizes the amount of services and software loaded on servers and workstations, which in turn minimizes the perceived risk to uptime introduced by a client agent. In this situation, the IT department would have to manually perform patch management on their “locked down” systems.

However, IT administrators also have the option to employ a client agent (using encrypted TCP/IP) on selected machines when it is desirable. Customers who have hundreds of systems that are not traditionally accessible via the network (including workstations that have been “hardened” for deployment on the Internet and machines located in untrusted environments), may wish to deploy PatchMatrix using this optional client agent. The client agent (also called Leaf Agent) is software installed on managed workstations and servers designed to do the work locally (with proper credentials) that one cannot perform remotely without the agent.

By giving IT managers the choice of installing agents on managed machines only when required, PatchMatrix minimizes the amount of risk to the enterprise overall and provides more control to the system administrator. With the agent, the workload is distributed to the managed machines – an excellent method for large numbers of systems and where bandwidth is an issue. Agents have many advantages, which are made apparent when considering the following:

- Not all systems are accessible without an agent.
- Resource management (bandwidth usage) is difficult without a distributed model, which requires an agent.
- Only the security layers provided by the OS manage security between the IT management software and the target systems.
- Secured collaboration of management tasks cannot be distributed without some agent-based architecture.
- Some systems have been hardened to prevent remote management.
- Some systems are accessible via a leased line, and the leased line is expensive when using excessive bandwidth. Simultaneously, an Internet connection may be available on the other side of this leased line. Therefore, an agent on those target machines may use the cheap Internet connection for downloading patches, thus saving the leased line's bandwidth.

Superior Validation Process

The PatchMatrix validation process checks that the information is correct for a set of installed patches and confirms that the process took place without any data corruption.

When determining patch status (whether a patch has been installed or not), PatchMatrix compares the file version of each file in the patch against files on the target computer(s). If these values match, the patch is considered installed. If the values do not match, most patch management tools stop there and report that the patch has not been installed. PatchMatrix takes it one step further. A patch is considered valid only if the installed file version on the target is greater than the one that is expected, the detected version matches another patch's file version from the

PatchMatrix patch management database, AND that other patch is installed on the target machine. This situation occurs when several patches happen to update the same file but don't necessarily supersede the patch before it, or if something causes the files for a patch to be restored to a previous state.

In addition, PatchMatrix validates all types of Microsoft patches, not just security patches. Authentium is committed to having the most comprehensive, independently tested meta-database of patches for the applications and operating systems supported by PatchMatrix.

Comprehensive Patch Management Database

PatchMatrix offers the industry's most comprehensive database of patching information. Its exclusive patch management database incorporates patches for Windows NT4/2000/XP/Server 2003, Terminal Server, IIS, SQL Server, Exchange, Internet Explorer, MDAC, Media Player, Windows Media Services, Net Meeting, Microsoft Office and Outlook.

Changes to the database are published three to five times per week. In addition, customers can add private hot-fixes simply by contacting Authentium's Technical Support. Optionally, customers can add their own **custom fixes** with validation information.

PatchMatrix's patch management database helps a company make informed decisions about what updates to deploy and their destinations. It inventories machines and fixes vulnerabilities in operating systems and applications based on the independent research and testing of Authentium's patch experts. As discussed in the next section, all possible combinations of deployments are tested to determine interdependencies and potential conflicts before adding them to the unique patch management database.

This database features:

- Categorized patches for technical and executive reporting and 'intelligence' on all vulnerabilities.
- Vulnerability and remediation information from leading sources.
- Extensive and ongoing patch management research on patch installation sequences and combinations.

-
- Continuing updates on released patches, including obsolescence and revision information.
 - Clear and concise information on patch installation sequences and combinations.
 - Exclusive set of instructions telling PatchMatrix how to deploy a patch as well as take inventory of the machine and validate the installation over time.

Independent Validation of Patch Interdependencies

Few companies can afford to devote the full-time efforts of their personnel to the testing of various software patch combinations. In recognition of this, a team of experts who research and test patches independently to ensure reliable software patch delivery and installation supports PatchMatrix's patch management meta-database. The analysis includes issues related to superseding patches, prerequisites and co-requisites of patches and coexistence of a patch for one application in a system with other applications.

PatchMatrix lowers risks by determining the proper sequence of deploying patches and includes warnings that notify administrators when a patch won't work with a system. If a patch has become obsolete or might cause a conflict, PatchMatrix will flag this, tell the user why and offer a remedy.

In short, PatchMatrix removes the guesswork and will not allow users to make deployment mistakes.

SUMMARY OF FEATURES

PATCHMATRIX FEATURES AND VALUE

In summary, Authentium PatchMatrix offers the following features and value:

- **Optional Agent Architecture:** An exclusive feature of PatchMatrix, the optional agent architecture saves time and overhead. System administrators need only deploy agents to machines that are isolated or locked down. All components, including agents, are secured using encrypted communications. Multiple management consoles enable sharing of management and reporting duties through a central hub.
- **Automated Discovery:** PatchMatrix checks periodically for new updates – including the latest security hot-fixes – via the Web. As they become available, they are turned over for immediate testing to Authentium’s test lab. Once tested, the program updates the local patch management database and displays the new updates. This feature enables an administrator to evaluate systems to find missing updates, track results and print reports. Profiles and user-defined groups can be used in combination with other search criteria during the discovery process. This makes it easy to manage machines by operating system, service pack level or patch level.
- **Automated Research & Testing:** Instead of having to manually research dozens of obscure warnings, some of which conflict with each other, PatchMatrix advises an organization about the fixes it can safely deploy based on their policies. This advice is based on PatchMatrix’s independently researched patch management database. In support of this, PatchMatrix provides articles online that answer questions about updates, including interdependencies, change history, deployment notes, reasons and category for the update, among others. Companies can include their own set of fixes in a customer fix database, complete with registry and file-based validations. Administrators have more information at their fingertips, which enhances their ability to make informed decisions and reduces errors. The Research View is organized in a tree hierarchy to organize patches effectively and speed the research process.
- **Inventory Taking:** PatchMatrix checks managed workstations and displays an easy-to-read inventory of the updates already installed on all systems. This quick inventory helps determine what actions are necessary for deploying additional updates. In contrast, a manual inventory could take as

long as ten or fifteen minutes per machine.

- **Policy Creation & Enforcement:** Administrators can determine the updates required for machines. Before, during, and after deployment, all managed systems can be measured against these policies with a Conformance Report. The report clearly pinpoints exceptions that must be attended to.
- **Automated Deployment:** PatchMatrix saves time by automating the process of managing updates remotely. Administrators can define a baseline of patches and deploy in a single step to all managed machines, or by groups of machines. PatchMatrix guides the process, informing administrators what patches are applicable and missing. It will not deploy a patch unless all interdependencies are addressed. The deployment can be scheduled at a later time, adding to the convenience of remote deployment. This feature alone saves tremendous time over manual approaches.
- **Patch Validation:** PatchMatrix provides information about the status of any scheduled deployment as well as installation validation. PatchMatrix's enhanced validation manages versions of files to enable it to recognize when a new patch has overwritten a file from an old patch.
- **Reporting:** This feature is used to match the baseline against the current inventory and manage by exception. Reports can be shared among management console users.
- **Support for Remote Machines:** PatchMatrix, which receives meta-database updates almost every day via the Internet, includes a Packed Update feature to support organizations with disconnected networks (e.g. military, government). This exclusive feature creates a portable update process between a connected workstation and the disconnected network to distribute the latest updates and maintain the same secure configuration across all networks.

CONCLUSIONS

Manual patch management is tedious and prone to error. It is also a costly and inefficient way to repair security holes. Even patching that is accomplished via custom scripting can take valuable hours to develop and test – time that a company's IT staff could spend on other critical system issues. In contrast,

Authentium's PatchMatrix offers a deployment process that reduces the work required from hours or days to minutes. It reduces costs and resources required for monitoring, assessing, managing and deploying fixes – not to mention the huge savings from prevention of security breaches.

PatchMatrix can be used with client agents installed or no agents simultaneously. Using agents grants the necessary access to manage systems in just about every case imaginable. But, not all systems require the use of agents. PatchMatrix offers both methods without forcing agents on IT managers who are sensitive about the subject of client agents. For this reason, it is very conceivable that companies may want to use both methods in the same implementation. This capability gives IT administrators the best of both worlds.

Unlike the Microsoft service, SUS, which gives the administrator little control over the process of patching, PatchMatrix offers the administrator full control. Remediation can be customized, for example, accomplished by department or by individual workstation or server or configured to begin after hours. In addition, PatchMatrix includes the ability to manage private hot-fixes. Customers can call Authentium and request an addition of their private hot-fix to the PatchMatrix patch management meta-database.

Moreover, PatchMatrix reduces the risk of deploying software updates because of its thorough approach to testing. Not only does PatchMatrix's patch management meta-database contain the industry's most comprehensive set of patches, it also contains the most reliable information. Every patch is fully tested for possible conflicts before it is added to the database.

By automatically performing fixes throughout the entire enterprise from a management console for a long list of Microsoft applications and operating systems, PatchMatrix saves time and resources, resulting in a tremendous ROI. Companies interested in calculating the ROI of automating their patch management process are invited to work through the example found in Appendix B of this paper.

In summary, PatchMatrix improves system security and saves time for IT professionals – while reducing costs and risks on behalf of the enterprise.

APPENDIX A:

PATCHMATRIX COMPONENTS

Authentium PatchMatrix 6.1's architecture consists of the following components,

- **Console:** This component is the GUI front-end for PatchMatrix. It is used to perform management functions and display network information via the Master Agent.
- **Master Agent:** Located on a designated machine, this component provides a central data repository for all Consoles as well as performing all network administration tasks. It can manage all selected machines via the same Windows RPC calls used in PatchMatrix v. 5.1. The Microsoft Scheduler service is no longer used, and has been replaced by our new Installer Service. The Master Agent also serves as a conduit for communications to Leaf Agents.
- **Leaf Agents (optional):** Leaf Agents are an optional component of PatchMatrix that perform all of the local functions required to manage a machine. By performing the tasks locally, it eliminates the need for any of the aforementioned ports or services. Leaf Agents communicate with the Master Agent via a secure TCP/IP connection. Leaf agents are recommended when a target machine is locked-down or isolated behind a firewall, and RPC is not a viable solution for remote management due to its unsecured nature. (While RPC performs authentication checks, it does not encrypt data transferred or protect it from tampering.)
- **Installer Service:** A robust substitute for the NT Scheduler service. Performs all patch installations.

These components communicate using a custom protocol for encryption and authentication, which is built from publicly reviewed cryptographic algorithms and NT Authentication. A single TCP/IP connection, established over user-configurable ports, is used for all communication between agents and the consoles. Communications to the Installer Service use local or remote Named Pipes as appropriate and also use a secure protocol.

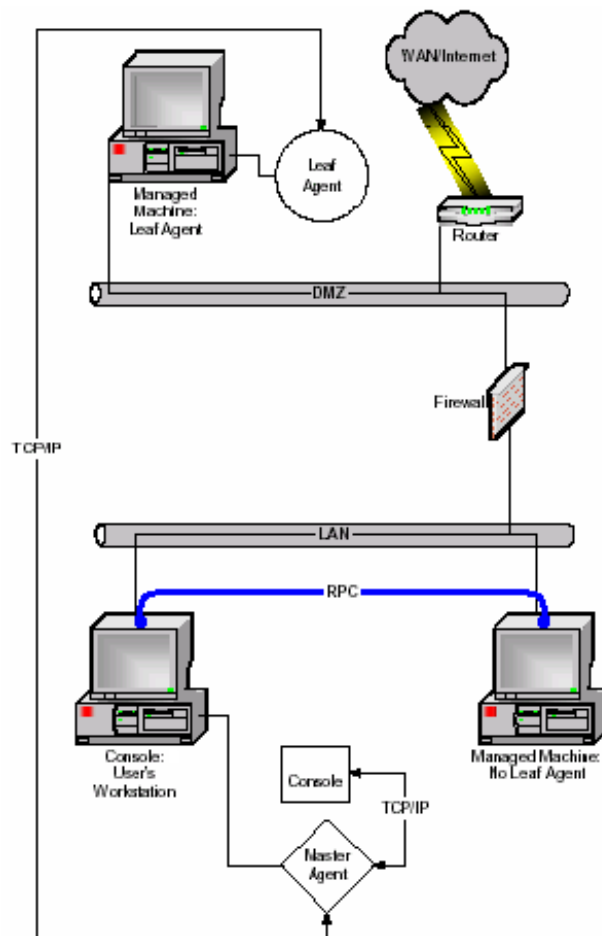
The agent architecture allows PatchMatrix to manage machines without the use of the RPC-based protocols (identified above), allowing it to work on networks where those protocols are disabled because of security concerns.

The agent architecture allows the network traffic to be minimized. Information about

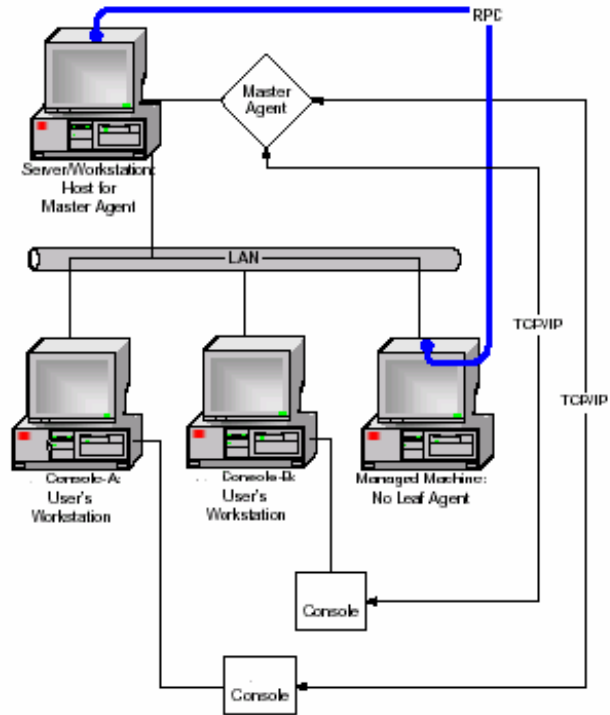
a machine is cached at the Master Agent. Leaf Agents detect when that information needs to be changed, and they update the Master Agent. Displaying this information on the PatchMatrix Console does not require it to be transmitted directly from the machine being managed.

PATCHMATRIX OPTIONAL AGENT ARCHITECTURE

The following illustration shows a common example of managing a system secured inside a DMZ. Thus, a system that is isolated can be managed, while simultaneously managing another system without installing an agent. Additionally, one can use both methods with virtually identical features and benefits.



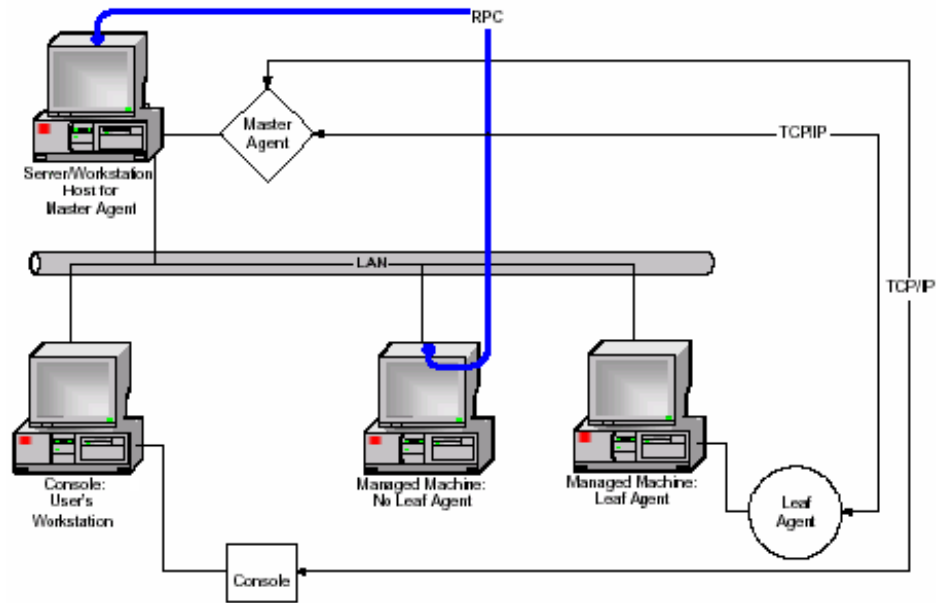
SAMPLE CONFIGURATION: MASTER AGENT PLUS MULTIPLE CONSOLES
The Master Agent and management console can be set up on the same device, or on different devices. The diagram shows a master agent and multiple consoles.



This configuration has multiple consoles. Any number of Consoles can be installed a multiple administrators to share the task of managing machines. Each Console interacts with the Master Agent and can display information about the network and each managed machine. Work done from one Console is visible from the other Consoles. The Consoles communicate with the Master Agent via a secure TCP/IP connection.

SAMPLE CONFIGURATION: MASTER AGENT WITH LEAF AGENTS

This configuration of PatchMatrix has a Console and a Master Agent on different machines. It also has a managed machine with a Leaf Agent and a managed machine without a Leaf Agent:



The Leaf Agent is installed remotely by using the File/Agent/Install Wizard menu item or by running the Agent Installation program locally on the target machine.

The Console communicates with the Master Agent via a secure TCP/IP connection. The Leaf Agent also communicates with the Master Agent via a secure TCP/IP connection. The Master Agent communicates with the managed machine that does not have a Leaf Agent using Windows RPC-based protocols.

APPENDIX B:

RETURN ON INVESTMENT CALCULATION

The ROI of PatchMatrix is determined by calculating the time it takes one to research, take inventory of all systems, deploy the updates and validate the installations. The following worksheet illustrates just one example of how PatchMatrix can save an organization time and money.

First, the hourly rate for an IT person is calculated. This is based on a fully burdened \$100K per year cost for that headcount. This produces an hourly rate for any time this person uses. This rate includes most overhead expenses. Now it is necessary to calculate the total time needed for just one update to all of the systems. In lines d, e and f, the time to research is included per update and the base-times are defined for inventory and deployment. This step calculates updating the systems manually.

Next, the example assumes that a firm has 100 systems. With these figures, it is possible to calculate the total time necessary for just one update to all of the 100 systems using a manual process. Line o shows that one update to all 100 systems costs 38h:36m or \$1,855. This time is used for research, inventory, deployment and installation validation of the updates. If one assumes that this task is performed only 4 times per year, the total cost comes to \$7,420 to manually update these 100 systems 4 times.

<i>Cost Factors:</i>		
a. Working hours in a year	2080 hours	52 weeks X 40 hours/week
b. Salary + benefits for one headcount	\$ 100,000	Typical burdened cost for one IT person
c. Calculated cost per hour for one headcount	\$ 48 /hour	= b ÷ a
<i>Time Factors:</i>		
d. Research and Discovery Time	4 hours	0 minutes
e. Time to update server	0 hours	45 minutes
f. Time to update workstation	0 hours	20 minutes
		Includes Web research, download times and programming
		Includes attended reboot
		Walk away for reboots
<i>Workload Factors:</i>		
g. # NT + W2K servers	3 servers	Servers running NT, W2K and Terminal Server
h. # NT + W2K workstations	97 wrkstns	Workstations running NT and W2K
<i>Calculation of Cost Savings:</i>		
i. # Hours research/download time	4.0 hours	= d
j. # Hours server updates	2.3 hours	= e X g
k. # Hours workstation updates	32.3 hours	= f X h
l. Cost for research/download	\$ 192	= i X c
m. Cost for server updates	\$ 108	= j X c
n. Cost for workstation updates	\$ 1,554	= k X c
o. Total cost for one series of updates	\$ 1,855	= l + m + n
<i>Other Factors:</i>		
p. Number of updates per year	4 updates/year	Number of times to implement hotfixes and service packs
q. Number of years subscription	1 year	Number of years for subscription of UpdateEXPERT
r. Extended Cost for updating for 1 year	\$ 7,420	= p X q X o

Try the Authentium **PatchMatrix ROI Calculator** at www.authentium.com.