Revista de Ciências Exatas e Tecnologia
Vol. III, Nº. 3, Ano 2008

UMA PROPOSTA DE GERENCIAMENTO E CONTROLE DE ATUALIZAÇÕES DE SEGURANÇA DE SISTEMAS OPERACIONAIS

Luis Henrique Medeiros da Rocha
IBM
lhmedeiros@yahoo.com.br

Rodrigo Xavier de Souza
IBM
rxavierx@yahoo.com.br

Rogers Vicco Paredes
Log & Print Gráfica e Logística S.A.
rogers_vp@yahoo.com.br

Wellington Campari
Interquattri Informática e Telecomunicações Ltda.
ton_campari@yahoo.com.br

Jeanne Dobgenski
Anhanguera Educacional S.A.
jeanne.dob@unianhanguera.edu.br

Anhanguera Educacional S.A.

Correspondência/Contato
Alameda Maria Tereza, 2000
Valinhos, São Paulo
CEP. 13.278-181
rc.ipade@unianhanguera.edu.br

Coordenação
Instituto de Pesquisas Aplicadas e Desenvolvimento Educacional - IPADE

Artigo Original
Recebido em: 15/7/2008
Avaliado em: 9/11/2008

Publicação: 8 de dezembro de 2008

RESUMO

Na tentativa de minimizar riscos de falhas de vulnerabilidade e segurança de sistemas operacionais os distribuidores das versões desses sistemas, periodicamente, disponibilizam atualizações conhecidas como *patches*. Entretanto, a quantidade de atualizações disponibilizadas é proporcional a quantidade de componentes do próprio sistema operacional, fato que torna a gerência de um ambiente mais onerosa à medida que aumenta a quantidade de servidores. Com a finalidade centralizar o controle dessas atualizações, reduzindo riscos de falhas humanas, oferecendo uma melhor visibilidade do status de atualização do ambiente ao administrador, fundamenta-se a proposta desse trabalho. Essas características foram disponibilizadas na ferramenta *TonTech - Patch Management*, a qual foi desenvolvida baseada em *software* livre, linguagens de código aberto conjugadas com o popular sistema gerenciador de banco de dados MySQL. O resultado é uma ferramenta adaptável e flexível para as necessidades particulares de cada ambiente, sem custos ao usuário interessado.

Palavras-Chave: Vulnerabilidade em sistemas operacionais, segurança em sistemas operacionais, gerenciamento de sistemas operacionais.

ABSTRACT

The attempt to minimize risks of operations system's vulnerability failures and safety, the distributors of these systems versions, periodically, available updates knows as patches. However, the quantity of the updates available it is proportional to the quantity of operating system components, fact that became more expensive due the servers quantity in an environment to be management. This work has an objective to centralize the control of that updates, reducing risks of human fails, offering a better visibility of updates status from the environment to the administrator. These characteristics were at tool *Tontech - Patch Management*, that was developed based on free software and open source languages conjugated with the popular database management system MySQL. The result is an adaptable and flexible tool to apply in particular needs of each environment, without high costs to the interested user.

Keywords: Operations system vulnerability, operations system security, operations system management.

1. INTRODUÇÃO

Com o objetivo de minimizar os riscos das vulnerabilidades existentes nos Sistemas Operacionais (SO) há um segmento de trabalho no campo de Tecnologia da Informação - TI, denominado de "*Patching Management*". Essa área busca gerenciar e aplicar soluções para corrigir tais vulnerabilidades, uma vez que as soluções são periodicamente disponibilizadas pelos distribuidores/fabricantes de *softwares*.

Como o ambiente acadêmico é o cenário propício para o surgimento de novas propostas de soluções, este trabalho visa apresentar a ferramenta "*TonTech - Patch Management*", que foi desenvolvida para gerenciar e automatizar as atualizações de sistemas operacionais Linux.

Influenciada pelos valores de contribuição à comunidade científica e o compartilhamento do conhecimento, a ferramenta foi desenvolvida sobre plataforma *open-source* de forma a poder atender ambientes de características distintas, facultando alteração de acordo com intenção de sua utilização.

2. REVISÃO DE LITERATURA

A pesquisa que fundamentou esse trabalho foi baseada em conceitos considerados pilares para a orientação de sua composição. Procurou-se referenciar aspectos clássicos do segmento, com abordagens sobre o que é o gerenciamento de *patch* e qual a devida importância a ser dada sobre todos os aspectos que circundam esse tema, como a segurança e os problemas que a envolvem em SO.

Macedo (2006) explica que a Microsoft devido à complexidade e ao grande volume de correções de seus sistemas computacionais desenvolveu o *Microsoft Solutions Framework* como meio de proporcionar uma proposta sólida, escalável e segura de atualização de seus produtos. A abordagem desta ferramenta tem como finalidade assegurar uma operação eficaz para empresas de todos os portes, tornando esta prática padronizada e consistente. Está composta por quatro etapas: avaliação, identificação, decisão e implantação.

Esse processo de gerenciamento de *patch* é uma tarefa multidisciplinar que envolve não apenas qualidades técnicas, mas principalmente visão de negócio. Embora cada etapa busque objetivos distintos, elas se completam facilitando a definição da es-

tratégia técnica que será utilizada no projeto, tais como a escolha da ferramenta de correção automatizada.

A segurança tem se tornado um dos principais focos no desenvolvimento de aplicações em geral (KROPIWIEC; GEUS, 2004). O crescimento do número de incidentes de segurança, entretanto, demonstra que os esforços estão sendo insuficientes para conter o avanço dos *hackers*. No trabalho desenvolvido por Kropiwiec e Geus (2004) são apresentados os paradigmas de segurança sobre os quais se baseiam os sistemas operacionais de uso mais comum e suas falhas. Buscam alertar os motivos que tem levado ao crescimento do número de ataques. Apresentam, também, novos paradigmas de segurança analisando os aspectos e dificuldades que impedem uma rápida adoção dos mesmos.

O trabalho de Mattos (2004) apresenta algumas reflexões acerca da origem dos problemas de segurança em sistemas operacionais. O trabalho defende a tese de que o tripé conceitual: virtualização do operador, virtualização do *hardware* e o conceito de programa - constitui a origem do problema. O trabalho indica que as pesquisas na área de sistemas SO baseados em conhecimento se apresentam como possibilidades reais a serem consideradas na busca das soluções para tais problemas.

Foi também realizada uma pesquisa de ferramentas já existentes e em utilização com o propósito de traçar uma linha comparativa em busca de um diferencial para o desenvolvimento do *TonTech - Patch Management*. Como base de referência, seguem alguns exemplos e suas principais características.

Kaseya Patch Management

O *Kaseya Patch Management* é uma solução, para manter os servidores, estações e computadores remotos atualizados com as últimas versões e *patches* de segurança. O gerenciador de atualizações *Kaseya* oferece uma auditoria automática de todos os *patches* e atualizações necessárias aos sistemas. Como administrador, tem-se todas as funcionalidades necessárias para automatizar todo o processo de atualização, incluindo a pesquisa, agendamento de pesquisa para cada máquina e quais *patches* ou atualizações estão aprovadas para a instalação. O resultado deste levantamento em cada computador é armazenado no servidor *Kaseya* para utilizações futuras. A partir da interface *web* para usuário, é visualizado o histórico completo de atualizações para cada máquina, incluindo as atualizações instaladas e não instaladas. Pode-se facilmente automatizar a distribuição e instalação de *patches* e atualizações em um agendamento pré-definido e

recorrente sem intervenção de administradores ou usuários. Após a finalização da pesquisa, poderá rapidamente revisar os resultados para cada máquina e decidir se, quando e como cada atualização ou *patch* será aplicada. Pode ser acessado de qualquer lugar e não exige um servidor especial ou reconfigurações da atual infra-estrutura de TI. Está disponível para os sistemas Windows.

Shavlik HFNetChkPro

Sua interface oferece controle completo de quais grupos serão testados, por qual critério, quando e como os *patches* serão aplicados. Basta configurar os parâmetros, clicar e instalar. O produto informará que os *patches* foram instalados. O HFNetChkPro reduz drasticamente tempos de administração, além de reduzir os riscos associados com vulnerabilidades de segurança. Permite aplicar *patches* de várias formas, de acordo com a criticidade, agrupamento de *patches*, tipos de *patch*, agrupamento de máquinas, modelos e outros. É também uma ferramenta de colaboração para compartilhar o conhecimento entre vários administradores. Automatiza a varredura de plataformas como Windows NT, XP, 2000, Windows Server 2003, Exchange, SQL Server, Outlook, Microsoft Office, Java Virtual Machine e outros. Baseado no padrão HFNetChk™, o qual é um mecanismo de varredura usado pela Microsoft em seu popular *Microsoft Baseline Security Analyzer* (MBSA). Ambos HFNetChk™ e MBSA foram desenvolvidos pela Shavlik Technologies. Usados mundialmente por empresas como a própria Microsoft e muitas outras empresas para garantir um gerenciamento completo de *patches* de segurança.

Altiris Patch Management

Com o *Altiris Patch Management Solution*, equipes de TI conseguem automatizar a instalação de *patches* lançados para corrigir, completar e atualizar as versões de *softwares* em uso nas corporações. A Altiris, líder em soluções de gerenciamento de infra-estrutura e do ciclo de vida de TI, oferece uma solução que pode auxiliar empresas a livrar seus ambientes de TI dos constantes ataques de vírus. Com essa ferramenta, departamentos de tecnologia das empresas podem acabar com as vulnerabilidades de sistemas operacionais e softwares de maneira prática e totalmente automatizada. Como os vírus exploram justamente os pontos fracos dos sistemas, os *patches* devem ser implementados o mais rapidamente possível. O *Patch Management Solution* faz uma varredura, de forma integrada ao MBSA, no sistema da empresa e compara o que há instalado nas máquinas com o conteúdo de um banco de dados *online* que acumula todas as vulnerabi-

lidades descobertas e seus respectivos *patches*. A solução tem recursos que providenciam a instalação das correções de forma automática, mas para isso o administrador da rede deve dar um comando autorizando a ação.

Prism Patch Manager

O *Prism Patch Manager* soluciona o problema de vulnerabilidade em sistemas operacionais, por meio da automação e simplificação do processo de gerenciamento dos *patches*. A solução localiza, efetua o *download* e instala os *patches* para manter o seu *software* atualizado, de forma rápida, fácil e confiável. A base de dados do *Prism Patch Manager* é administrada por uma equipe de especialistas que procura e testa todos os *patches*. Informações detalhadas sobre dependências e pré-requisitos estão sempre à mão, para que sejam evitados conflitos de *patches* e problemas de utilização. O produto gerencia *patches* de Windows, Linux e Solaris e mais uma abrangente gama de aplicativos da Microsoft.

3. SEGURANÇA EM SISTEMAS OPERACIONAIS

Considerando-se os objetivos deste trabalho, alguns conceitos fundamentais devem ser analisados antes de se abordar a solução proposta. Assim sendo, esta seção explora conceitos básicos e faz algumas considerações. As questões a serem apresentadas são relativas à segurança e falhas de sistemas operacionais, bem como uma explanação sobre atualizações.

De acordo com Tanenbaum e Furmankiewing (2006), os sistemas operacionais surgiram com dois objetivos principais: criar uma camada de abstração entre o *hardware* e as aplicações, e gerenciar os recursos de maneira eficiente e transparente ao usuário. Os SOs são *softwares* e estão sujeitos às falhas de implementação como qualquer outro, sendo que essas ocorrem dentro do próprio sistema, devido aos erros na implementação causadas por eventuais deslizes na fase do projeto ou na fase de transição do projeto para a codificação.

3.1. Atualizações de segurança

Os SOs são carentes de atualizações contra vulnerabilidades de segurança e falhas de funcionamento. Uma atualização de segurança é uma correção amplamente difundida para algum tipo de problema de segurança que pode ser usado na exploração de um

sistema em termos de vulnerabilidades ou falhas de funcionamento. A categoria de atualizações de segurança mais crítica é denominada vulnerabilidade do "Dia Zero", ou seja, o desenvolvedor não soube de sua existência até que um invasor a utilizou. Felizmente, esse tipo é raro. Uma outra categoria compreende problemas que são publicados em algum fórum aberto antes do fabricante ser notificado. Isso deu origem ao que se chama de "divulgação responsável", pela qual o desenvolvedor tem a chance de corrigir o problema antes que se torne público. No entanto, alguns analistas de segurança de aplicativos, seja qual for o motivo, decidem não seguir essa conduta de bom comportamento. Baseado nas experiências dos autores do presente trabalho, pode-se afirmar que em ambiente profissional o ato de não seguir tal conduta coloca os consumidores em risco e faz com que o fabricante se apresse para produzir atualizações de segurança, os quais não terão a mesma qualidade de um *Service Pack* desenvolvido e testado minuciosamente. A maioria das atualizações de segurança que foi recolhida e relançada se enquadra nessa categoria. Diferentemente, uma terceira categoria de atualizações de segurança é aquela em que o usuário relata para o fabricante um problema encontrado, aguardando o tempo necessário para corrigi-lo de forma adequada, antes de divulgá-lo ao público.

Uma variante visivelmente ausente dessa lista é a correção de vulnerabilidades encontradas pelo desenvolvedor. Geralmente essas vulnerabilidades, descobertas internamente, são encontradas durante a criação de uma atualização de segurança para o mesmo componente. Em algumas vulnerabilidades descobertas pelo fabricante a correção aguarda um *Service Pack* que, em geral, são submetidos a rotinas de testes mais elaboradas.

Concluindo, todos os sistemas precisam de atualizações de segurança de alguma forma e gerenciá-las é uma necessidade. Isso não significa que gerenciar *patches* seja uma experiência simples e prazerosa, mas uma forma de aumentar a confiabilidade no sistema e a sua eficiência, demandando menos tempo no processo de atualização e reduzindo custos operacionais. A segurança é um processo contínuo.

3.2. Falhas de segurança

Considerando o conhecimento e experiências passadas com os autores desse trabalho, observa-se que a detecção de falhas na segurança é o ponto mais complexo na defesa de SO. Trata-se por falha de segurança qualquer acesso a sistemas e informações que não seja expressamente autorizado, nem mesmo realizado de forma clara - obedecendo

às normas de conduta dos protocolos de redes definidos - ou ainda, não estejam de acordo com as regras estabelecidas dentro da instituição proprietária das informações. É também estabelecida como falha a interrupção dos serviços por motivos não definidos, causadas por pessoas ou máquinas não autorizadas, de forma comum ou não.

O acesso aos serviços por meio de suas falhas ou vulnerabilidades é visto como falta grave e exige atenção especial, pois nestes casos são necessárias as interrupções dos serviços. Os erros em protocolos de comunicação têm que ser monitorados exaustivamente, pois não é possível a interrupção do uso do protocolo de comunicação no sistema, sendo assim, as falhas devem ser isoladas e resolvidas com a máxima prioridade. Em função de falhas nos protocolos é que são criados os ataques e as invasões mais eficientes, sendo quase imperceptíveis aos dispositivos de segurança que são construídos baseados nestes protocolos. Para estes ataques, o estudo e a análise do comportamento do sistema levam à conclusão que algo está errado, mal intencionado ou é destrutivo.

É importante para a segurança dos sistemas operacionais identificar, dentro das requisições de serviços e pacotes de dados, quais destes são corretos, incorretos, alarmes falsos ou mecanismos de distração. Um meio de levar um sistema à falhas e, conseqüentemente, ficar vulnerável, é sobrecarregá-lo com falsos ataques rápidos e simples, desviando a atenção para causas menores e enfraquecendo pontos vitais que podem ser ou estão sendo atacados.

Dessa forma, a aplicação dos *patches* de segurança é uma ação de grande importância, uma vez que eles são disponibilizados no intuito de sanar as vulnerabilidades supracitadas.

4. UMA PROPOSTA PARA GERENCIAMENTO E CONTROLE DE ATUALIZAÇÕES DE SEGURANÇA

Esta seção apresenta o desenvolvimento da proposta de uma solução alternativa que gerencia e controla *patches* para a atualização e a segurança em sistemas operacionais Linux.

Essa proposta busca uma solução eficiente para os problemas apresentados anteriormente. Para isso, procura agregar conhecimentos desenvolvidos em outras áreas de pesquisa, particularmente em sistemas operacionais, banco de dados, redes de computadores e linguagem PHP.

4.1. Tecnologias adotadas

O planejamento de um sistema de gerenciamento de *patches* de segurança para plataforma Linux é viável devido a sua grande utilização como SO de servidores em inúmeras empresas nos seus *datacenters*. A escolha da utilização de uma distribuição cujo padrão de pacotes componentes do sistema seja do tipo RPM (*Red Hat Package Manager*) se fortaleceu devido ao fato da principal distribuição comercial Linux adotada pelas empresas ser a *Red Hat Enterprise*. Isso atrai um alto índice de investimento em pesquisas sobre os mesmos, tornando este padrão mais confiável. Além disso, poderá ser incorporado de forma positiva em outras distribuições Linux.

Devido a tais características, optou-se por utilizar a distribuição Fedora como base de pesquisa e testes, a qual é uma versão alternativa e *freeware* desenvolvida pela *Red Hat.*

No âmbito dos *softwares* de código aberto e de livre utilização, como opção ao controle da base de dados necessária para as funcionalidade da solução, adotou-se o sistema gerenciador de banco de dados (SGBD) MySQL. Oferece todas as funcionalidades dos produtos comerciais como rapidez, execução multitarefa e multiusuário.

Em virtude da característica de facilidade de acesso, navegação e visualização, a *interface* da ferramenta desenvolvida, foi idealizada para operação por meio de um navegador *web*. Para que o serviço apresente funcionalidade, o mesmo deve ser mantido por um servidor *web* de maneira que disponibilize informações, como as páginas do produto. O servidor *Web* utilizado para desenvolvimento e testes foi o Apache, pois possui diversas características como ser configurável, possuir alta performance e robustez. Quanto ao desenvolvimento do código, elegeu-se a linguagem de programação PHP que permite a criação de sites dinâmicos. Ou seja, favorece a interação direta com o usuário. Outra característica fundamental do PHP se evidencia no quesito segurança, pois o código da aplicação reside no servidor, impossibilitando que os usuários o acessem. Isso é muito importante porque dados confidenciais como usuários, senhas, dentre outras informações, são componentes do código. Mais vantagens a serem destacadas são relacionadas à integração desta linguagem com banco de dados e o fato de possuir licença pública geral (GPL - *General Public License*) para utilização.

As conexões com tráfego de dados entre os servidores são realizadas sobre o protocolo SSH (*Secure Shell*), que garante a segurança da transação porque implementa tunelamento protegido por criptografia. A utilização deste se mostrou viável devido à

integração de seus recursos de conectividade segura e execução de comandos remotos, que são facilmente integrados com a linguagem PHP.

Finalizando o conjunto de tecnologias adotadas, deve-se citar o uso de uma linguagem de *scripts* de SO, *shell-script*, utilizada para integrar recursos providos pelas funcionalidades das rotinas do próprio sistema operacional, integrando-as ao aplicativo PHP.

4.2. Solução proposta

O *TonTech - Patch Management* propõe uma solução alternativa para gerenciamento e controle de *patches* de atualização e segurança em sistemas operacionais Linux (Figura 1). É passível de adaptação a diferentes características de ambiente, de forma a viabilizar o gerenciamento e controle de diferentes distribuições.

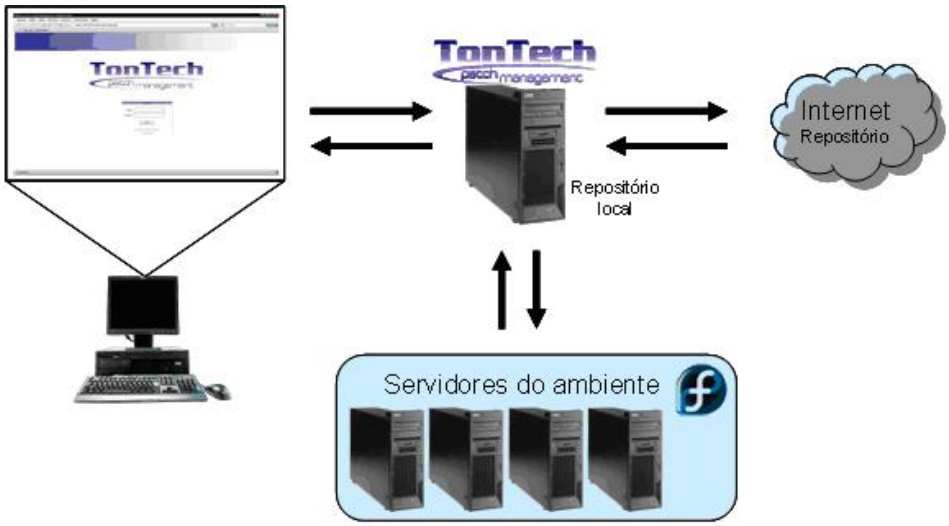

Figura 1. Ambiente da solução proposta.

Tal mecanismo facilita a obtenção das informações relativas aos estados de atualização do parque de servidores, minimizando o tempo que seria empregado para a realização desta tarefa caso fosse realizada manualmente. O relatório proporcionado pela ferramenta permite uma análise ágil e clara, ajudando na adoção de uma estratégia de trabalho cujo objetivo seja mitigar as vulnerabilidades de segurança do ambiente de forma mais eficaz.

O projeto utilizou uma estrutura de interconexão entre os servidores, os quais estavam em uma rede privada. Eles podem representar servidores de banco de dados, de aplicações, dentre outras características usuais dos modelos *datacenters* das empresas.

Um outro servidor provido de conexão com a Internet foi usado como base para o *TonTech - Patch Management*, e exerce as tarefas de concentrador dos inventários dos pacotes vigentes nos servidores do ambiente. Munido destas informações, este servidor coleta as informações dos pacotes disponibilizados como atuais, no site do distribuidor do SO, possibilitando a visualização do estado real de atualização dos servidores do ambiente, em relação ao estado mais atual disponível.

A estrutura das tabelas criadas que compõem a base para armazenamento dos dados da ferramenta conta com cinco tabelas as quais foram intituladas como: "pkg_server", "pkg_site", "server_hostname", "so" e "users", as quais estão relacionadas conforme indicado na Figura 2.

A tabela "server_hostname" concentra o armazenamento dos dados relativos aos servidores, sendo estes o "hostname" (único), o "ip" (único) e o código do SO utilizado pelo servidor, tal código estará vinculado a tabela de SOs, vale lembrar que para essa tabela há o campo "hostname" como chave - primaria, pois cada servidor será identificado por esse dado, sendo também esse campo utilizado para vincular a "server_hostname" com a "pkg_server".

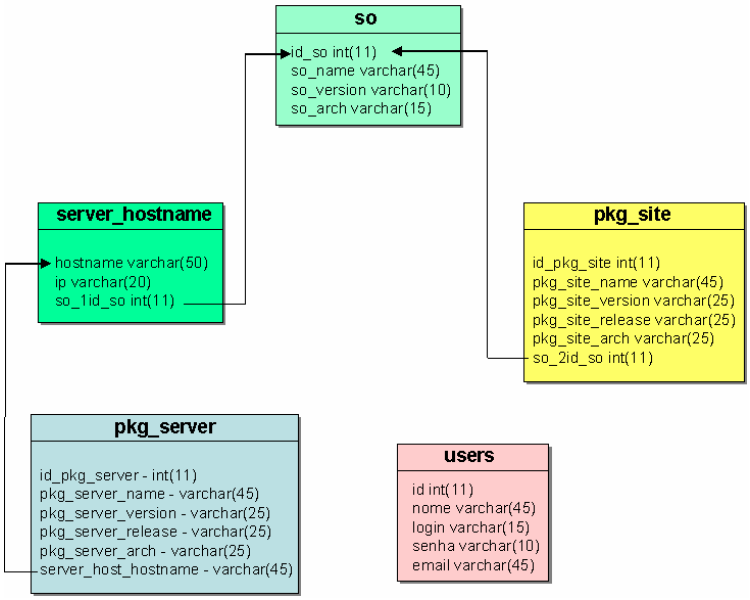

Figura 2. Modelo do banco de dados da solução.

Na tabela "so" os dados para cada cadastro do sistema operacional são: nome, versão e plataforma: sendo gerado um "id" para cada cadastro o qual servirá para vincular a tabela "so" com "server_hostname" e com "pkg_site".

A principal característica da tabela "pkg_server" é o armazenamento de uma grande quantidade de dados, já que na mesma estarão concentrados os registros dos

pacotes dos servidores. Oferecerá informações como: "id", um nome do pacote, uma versão do pacote, um release do pacote, uma plataforma do pacote e, por final, um campo que indicará a qual servidor o pacote pertence sendo ele o campo "server_host_hostname".

A tabela "pkg_site" é semelhante à "pkg_server", mas indicará a qual sistema operacional o pacote pertence através do campo o "so_2id_so".

Finalmente, a tabela "users" reunirá os cadastros dos usuários. Para cada cadastro será necessário um *login* (único), senha, nome do usuário, *e-mail* que será utilizado para o envio de informações do cadastro e recuperação de dados. Por fim, será gerada uma "id" para a conta.

4.3. Mecanismos de funcionamento desenvolvidos

Com o objetivo de detalhar as funcionalidades e características do funcionamento do resultado do projeto, são descritas a seguir os módulos que combinados compõem a solução.

Obtenção do Inventário dos Servidores do Ambiente

A ferramenta conta com uma classe de funções particular da linguagem PHP, identificada como *ssh2*, que é parametrizada por valores como endereço IP do servidor, *login* e *password*. Além disso, provê a dinâmica de conexão remota e tráfego das informações entre o servidor do *TonTech - Patch Management* e seus clientes cadastrados. E, para isso, usa as funcionalidades e segurança de tunelamento de transação de dados providos pelo protocolo SSH.

Nesse sentido, a ferramenta se conecta remotamente no servidor especificado, executando a geração do inventário dos pacotes instalados concentrando tais informações em um arquivo específico. Na seqüência, a ferramenta executa a cópia desse arquivo para o servidor base do *TonTech - Patch Management,* no qual tais dados serão carregados automaticamente em uma tabela do banco de dados que fora criada para suportar tal estrutura de dados. Tal processo é executado periodicamente de acordo com o agendamento do administrador do SO.

Concentração dos pacotes para instalação

A ferramenta realiza uma conexão diretamente ao repositório de pacotes nas fontes disponibilizadas pelos distribuidores das versões do SO, executando o *download* dos pacotes para uma área local - que pode até mesmo ser o servidor base do *TonTech Patch Management*. Tem a finalidade de concentrá-los para minimizar eventuais elevações de tráfego de dados na rede, e favorecer uma consulta mais centralizada dos índices dos pacotes tidos como mais atuais.

Conforme mencionado, o processo é executado via agendamento do administrador do SO, bastando a inclusão da chamada do *script* no agendador de tarefas do mesmo (*cron*). A garantia de correspondência entre fonte e repositório local é dada por meio da sincronização proposta pela estrutura de consolidação de repositório local.

Obtenção das referências de pacotes atuais do repositório

Utilizando como base os pacotes presentes no repositório local, de acordo com a versão do SO desejado, a ferramenta executa uma consulta e cria uma lista com as informações relativas aos pacotes oferecidos no repositório. Dessa forma, inclui automaticamente as últimas versões dos pacotes disponíveis em uma tabela relativa ao armazenamento de tais informações no banco de dados.

Engajado nesse processo ocorre a comparação do inventário de pacotes instalados nos servidores com os pacotes disponibilizados no repositório do sistema operacional. Tem por objetivo analisar se os pacotes instalados no servidor estão atualizados, caso as versões comparadas sejam iguais.

Atualização dos pacotes

As atualizações dos pacotes selecionados na *interface* da ferramenta são viabilizadas novamente pela dinâmica entre o poder provido pela linguagem PHP e funcionalidades do protocolo SSH. Isso permite que os servidores alvos de atualização se conectem no repositório local e realizem a atualização dos pacotes selecionados, o que ocorre mediante a execução do código descrito na Figura 3.

```
//cria conexao com o servidor
if (!($conexao = @ssh2_connect($linha['ip'], 22))) {
  echo "ERRO -> Conex&atilde;o com servidor ".$linha['hostname']." falhou!<br>";

} else {
  //valida conexao com login e senha
  if (ssh2_auth_password($conexao, $loginssh, $senhassh)) {
          echo     "Conex&atilde;o     e     Autentica&ccedil;&atilde;o     com     servidor
".$linha['hostname']." realizada com sucesso!<br>";
          //executa comando para criar arquivo com a lista dos pacotes instalados no ser-
vidor
```

```
          if (ssh2_exec($conexao, "yum -y update $lista_pacotes")) {
                  echo "Atualiza&ccedil;&atilde;o realizada com sucesso!<br>";
          } else {
                  die('Execu&ccedil;&atilde;o do comando de atualiza&ccedil;&atilde;o fa-
lhou!<br>');
          }
  } else {
          die("ERRO -> Autentica&ccedil;&atilde;o com servidor ".$linha['hostname']." fa-
lhou!<br>");
  }
}
```

Figura 3. Código que possibilita a atualização dos pacotes selecionados.

4.4. Aspectos do funcionamento

A *interface* da ferramenta se apresenta de forma simples e funcional. Tal fator foi pontuado na análise prévia ao desenvolvimento, uma vez que a identificação do usuário com a ferramenta minimiza a resistência de sua utilização, permitindo o reconhecimento do valor que ela agrega as suas rotinas.

Desta forma, a título de referência, serão descritas algumas características pertinentes às janelas da ferramenta produzida.

O *TonTech - Patch Management* é iniciado a partir de uma requisição via navegador (*browser)* exibindo a tela na qual é efetuado o processo de *login* do usuário para o acesso ao produto - que possui o gerenciamento de cadastros de usuários (Figura 4). Caso o acesso não seja validado é exibida uma notificação (Figura 5).

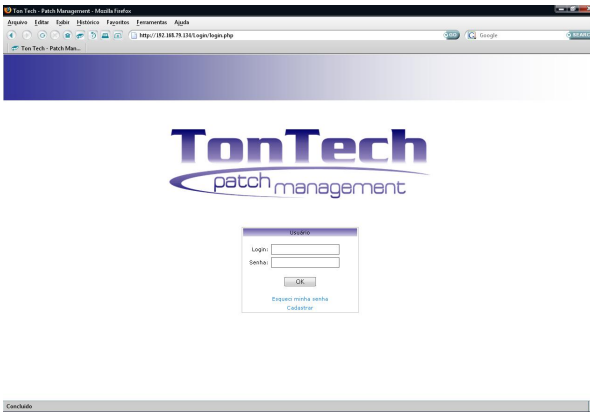

Figura 4. Acesso ao *TonTech - Patch Management.*

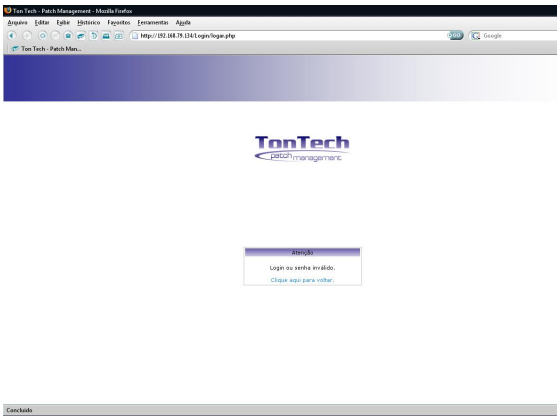

Figura 5. Validação de acesso não realizada.

É importante ressaltar que a segurança do acesso também é reforçada por uma restrição quanto ao cadastro, sendo permitido realizar tal processo apenas com a utilização de uma chave de autenticação específica fornecida apenas ao candidato.

Validado o acesso, o ambiente de navegação é constituído de um menu lateral com as opções de utilização da ferramenta. Essas opções são fixas, ou seja, estarão presentes em todas as telas selecionadas no menu.

A tela inicial é o "Informativo" que apresenta as informações sobre a ferramenta (Figura 6). No menu "Recados" é possível inserir e visualizar recados de caráter informativo referente ao ambiente (Figura 7). Por exemplo, pode-se informar que um determinado grupo de servidores ainda não foi verificado, ou que a inclusão de servidores ainda está pendente, até mesmo informar possíveis problemas técnicos em determinados servidores entre outros.

Fica reservado para exibição máxima de 10 ocorrências, e a partir da inclusão do 11º, o último registro da lista é removido, dessa forma os recados mais antigos serão eliminados permanentemente. Por questões de boas práticas de codificação, foi aplicado um filtro que inibe o funcionamento do PHP dentro das páginas de recados criadas pelo usuário, dessa forma os comandos do PHP utilizados dentro delas serão reconhecidos apenas como textos normais.

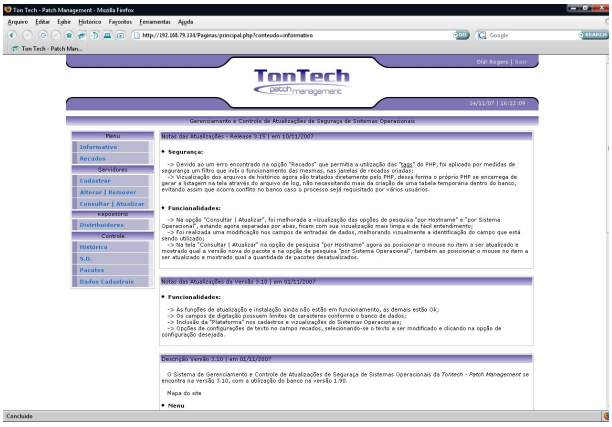

Figura 6. Página inicial da ferramenta.

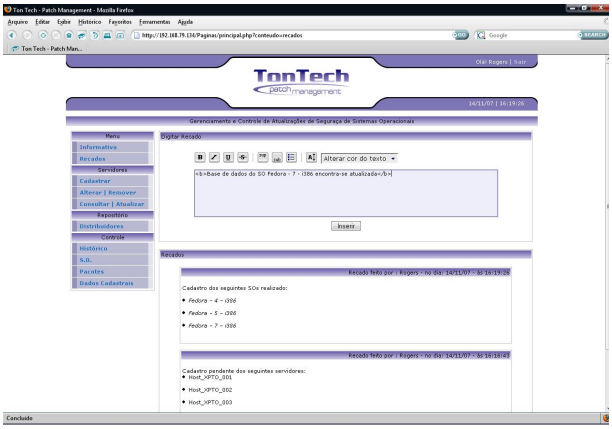

Figura 7. Recados.

Em "Servidores" há opções relacionadas diretamente à administração dos servidores do ambiente.

Como primeira escolha tem-se o botão "Cadastrar", que corresponde a tela de cadastro de servidores, tal processo se dá por meio das entradas de informações da máquina a ser cadastrada, como o HOSTNAME, o IP, o SO, ressaltando que os dados fornecidos só serão aceitos para cadastro, caso sejam únicos. Isto significa que não poderão ser cadastrados dois servidores com o mesmo HOSTNAME ou com o mesmo IP. Além desses dados base, tem-se mais uma opção oferecida: a "carregar pacotes" - consiste em carregar o inventário dos pacotes instalados desse servidor na base de dados.

Mas, essa tarefa pode ser feita *a posteriori* na opção "Pacotes". Esse recurso é importante para observar a situação do servidor a ser cadastrado, o qual pode estar ainda, desconectado da rede.

Caso a opção escolhida seja a de fazer o carregamento do inventário, a ferramenta realizará uma conexão com o servidor. Utilizará o IP fornecido e gerará um arquivo na máquina acessada com inventário dos pacotes instalados, cujo arquivo é copiado para o servidor hospedeiro da ferramenta e os dados são carregados na base de dados. Existem também as opções "Alterar | Remover".

Na terceira opção desta sessão (Servidores), tem-se a utilização do principal processo da ferramenta da *TonTech - Patch Management*, que provê a possibilidade de atualização dos pacotes de um determinado servidor ou de um grupo de servidores, sendo o acesso executado pelo botão "Consultar | Atualizar". O processo realizado nessa área tem como objetivo a verificação dos pacotes dos servidores - se estão desatualizados, permite a atualização, que pode ser executada por pacote específico ou em todos os pacotes de um conjunto de servidores - ao mesmo tempo. Disponibiliza-se também a geração de um histórico do inventário presente na base de dados antes da realização da atualização. Existem dois critérios a serem utilizados.

O primeiro é a pesquisa por "Hostname" que visa à exibição do inventário dos pacotes presente na base de dados de um servidor selecionado. São realizadas comparações entre os pacotes do servidor e os pacotes dos distribuidores, relacionando-os por suas versões de SO respectivas. Identifica-se quais são os pacotes que estão desatualizados e quais não estão instalados, fornecendo três formas possíveis de exibição, apresentados a seguir.

- **Pacotes instalados:** lista todos os pacotes instalados no servidor e os pacotes que estão desatualizados, disponibilizando uma opção para atualizá-los. Caso ocorra uma solicitação de atualização a ferramenta gera um histórico atual da base de dados com a relação dos pacotes instalados e, em seguida, realiza uma conexão como servidor selecionado. Envia comandos específicos do SO com a relação dos pacotes para a execução da atualização, exibindo em seguida na tela a situação do processo.
- **Pacotes desatualizados:** lista apenas os pacotes que estão desatualizados, disponibilizando uma opção para atualizá-los. Se ocorrer uma solicitação de atualização e se for realizado o mesmo processo descrito, retorna-se, na tela, as informações do processo.

- **Pacotes não instalados:** lista todos os pacotes que não estão instalados no servidor, porém, disponíveis no repositório. Logo, fornece uma opção que permite instalá-los. Após a solicitação da instalação a ferramenta realiza os passos iniciais do processo de atualização, diferenciando-se desse pelos comandos do SO a serem executados internamente. Exibe ao final a tela da situação do processo.

O segundo critério é a pesquisa "por Sistema Operacional". Esse item tem por finalidade listar todos os servidores com um mesmo tipo de SO previamente selecionado. Dessa forma, junto com a exibição dos respectivos servidores é mostrada a situação dos mesmos com relação aos níveis de atualização dos pacotes, identificando um servidor como atualizado ou desatualizado (para um ou mais pacotes desatualizados). Há a opção de realizar a atualização de um ou vários servidores.

Assim como nos demais processos descritos, este realizará a criação do histórico e, em seguida, a conexão aos servidores selecionados, sendo, por fim, exibida na tela as informações da situação desse processo.

Em "Repositório", encontra-se disponível a consulta dos pacotes fornecidos pelos distribuidores dos SO. A listagem gerada é baseada nos arquivos existentes no repositório local. Dessa forma, ao selecionar um determinado SO são exibidos na tela todos os pacotes disponíveis para a versão escolhida, oferecendo uma referência para comparações entre as versões e *releases* dos pacotes novos.

Finalizando este conjunto de opções da ferramenta, encontra-se a sessão "Controle". Análogo a primeira opção, tem-se disponível a visualização da listagem dos arquivos de históricos. Esse arquivo é gerado durante as alterações dos pacotes (seja por atualização ou por instalação). Ao ser escolhido um determinado registro de ocorrência, o mesmo é carregado na tela a fim de prover uma referência que poderá ser utilizada para verificação do estado dos servidores antes das alterações.

O processo de administração dos SOs permite visualizar todos os sistemas operacionais cadastrados na base de dados, incluir ou remover registros. Por medidas de segurança não é permitido remover um registro se estiver vinculado a algum servidor. Para evitar redundância não é possível realizar dois cadastros se os dados da versão e da plataforma forem iguais ao que se deseja incluir.

Outra opção a ser explorada é "Pacotes" que fornece a funcionalidade de atualização da base de dados com o inventário dos pacotes dos servidores ou com o inven-

tário dos distribuidores. Estas duas escolhas realizadas são para servidores ou distribuidores.

- **Para servidores:** atualiza a base de dados com o inventário dos pacotes atuais instalados no servidor. Ou seja, após ser realizada a atualização de determinados pacotes, elas não serão apontadas na listagem apresentada pela ferramenta. Isso será realizado após atualizar a base, conectando-se ao servidor, utilizando o IP cadastrado, gerando um arquivo na máquina acessada com inventário dos pacotes instalados. Em seguida o arquivo é copiado para a máquina em que se encontra a ferramenta, sendo o inventário antigo eliminado da base de dados e o novo arquivo gerado e carregado nessa base. Esse processo pode ser realizado em mais de um servidor, sendo exibido um relatório indicativo se o processo foi realizado com êxito ou se ocorreu alguma falha.
- **Para distribuidores:** atualiza a base de dados com o inventário atual do repositório. É feita uma varredura no repositório local e é gerado um conjunto de arquivos (baseados nos pacotes dos tipos: base, extras e *updates*) dos SOs selecionados. Após a criação dos arquivos o inventário antigo é eliminado da base de dados e, em seguida, serão carregados na base os arquivos de "base" e "extras". Na seqüência é iniciado o carregamento do arquivo "updates", verificando a existência do pacote e, caso isso ocorra, ele será substituído pelo nome do pacote que está no arquivo "update". Esse processo poderá ser realizado em mais de uma versão de SO.

Por fim, "Dados Cadastrais", que viabiliza a manutenção da conta do usuário e permite alterações na senha e/ou no e-mail cadastrados. A cada atualização a ferramenta envia, automaticamente, um e-mail com os novos dados ao usuário.

4.5. Vantagens da solução proposta

A utilização do *TonTech - Patch Management* se mostra viável, principalmente, por oferecer a centralização das informações relativas aos servidores. Pois, além dos inventários de pacotes vigentes, tem-se controle de endereço IP e versão de SO, em um relatório ou nas seções do aplicativo.

Outro grande atrativo é a redução de *workload*[1] para a execução da tarefa por parte do administrador do ambiente, em relação à realização convencional de tal procedimento. Isto permite que o tempo do administrador do ambiente seja direcionado

[1] Volume de carga de trabalho.

para atividades preventivas, ao invés de reativas, indicando economia. Esta característica é acrescida, ainda, pela possibilidade de acesso remoto, pois o acesso ao produto é realizado via *browser*, não restringindo a localização física do administrador.

Considerando um ambiente de médio/grande porte o *TonTech - Patch Management* provê confiabilidade na obtenção e análise dos dados, reduzindo a possibilidade de falhas humanas.

Em relação aos demais *softwares* similares já existentes, o *TonTech - Patch Management* oferece sua implantação isenta de custos já que sua distribuição é gratuita, por meio da GPL (Tabela 1).

Tabela 1. Comparação entre algumas ferramentas disponíveis.

Produto	Preço[2]
FNetChkPro (Shavlik)	116 dólares (aquisição para cinco micros)
Kaseya Patch Management	Não divulgado
Prism Patch Manager (Prism)	912 dólares (aquisição para 25 micros)
Altiris Patch Management (Symantec)	30 dólares por micro

É ainda passível de adaptabilidade para outras versões de sistemas operacionais e outras funcionalidades, pois a ferramenta foi desenvolvida com o intuito de ser de caráter *open-source* e poderá ser disponibilizada em sites de hospedagem de projetos com essa mesma característica.

5. ANÁLISES E RESULTADOS

Para efeito de testes, em virtude da indisponibilidade de submeter servidores de um ambiente de produção de um *datacenter* ao uso da ferramenta desenvolvida, procurou-se simular 20 servidores virtuais distribuídos em quatro computadores pessoais, por meio de *software* de virtualização. Estes interligados por um roteador, permitindo a interconexão entre os mesmos.

Os computadores possuíam características de *hardware* distintas, porém, uniformes quanto à quantidade de memória RAM, de forma a não degradar consideravelmente o desempenho dos servidores virtuais (Tabela 2).

[2] Valores aproximados.

Tabela 2. Configuração das máquinas utilizadas nos testes.

Processador	Disco Rígido	Memória RAM	Placa de Vídeo
Intel Pentium 4 3.2GHz HT	80GB	2 GB DDR2 533MHz	GeFORCE EN6600 GT
AMD Sempron 3100+ S462	80GB	2 GB DDR 400MHz	GeFORCE FX5200
Intel Pentium D 925 3.0GHz	250GB	2GB DDR2 667MHz	GeFORCE 7300GT
AMD Sempron 2800+ S754	80GB	2GB DDR 400MHz	ATI X300

Houve a necessidade da configuração de um repositório local no servidor da aplicação, contendo os pacotes necessários para viabilizar as atualizações dos servidores testados. Fase essa que, inicialmente, requer tempo e banda de conexão de Internet para possibilitar a disponibilização local de todo o conteúdo.

Foi explorada a funcionalidade do gerenciamento de controle de inventário dos pacotes instalados nos servidores, fator que é de grande ônus ao administrador de TI, principalmente quanto a concentrar as informações de forma prática para a consulta e visualização.

Logo durante os testes, dedicou-se atenção especial à análise do controle do inventário, principalmente para a consolidação de um relatório do sobre *status* anterior e posterior ao processo de atualização, valendo-se da funcionalidade da própria ferramenta.

Foram realizados testes de atualização com o uso da ferramenta e de forma manual, no intuito de comparar e visualizar a real economia de tempo utilizando o *TonTech - Patch Management*. As atualizações foram feitas por três indivíduos A, B e C. Cada atualização foi repetida cinco vezes, sendo a média apresentada na Tabela 3.

Tabela 3. Comparativo entre a atualização manual e com a ferramenta.

Tempo gasto com as atualizações (min.)								
Qtde. Servidor	**Atualização Manual**				**Atualização com *TonTech***			
	indivíduo				indivíduo			
	A	B	C	**Média**	A	B	C	**Média**
1	122,4	120,0	117,2	**119,9**	56,8	56,0	57,4	**56,7**
10	676,5	660,0	644,6	**660,4**	87,6	87,4	87,8	**87,6**
20	1353,0	1320,0	1289,2	**1320,7**	103,2	105,2	103,4	**103,9**

Observa-se que o tempo de atualização (feito manualmente) aumenta muito à medida que são atualizados mais servidores. Porém, com a ferramenta *TonTech* o aumento não é tão grande, porque esta realiza a atualização dos servidores simultaneamente (Figura 8).

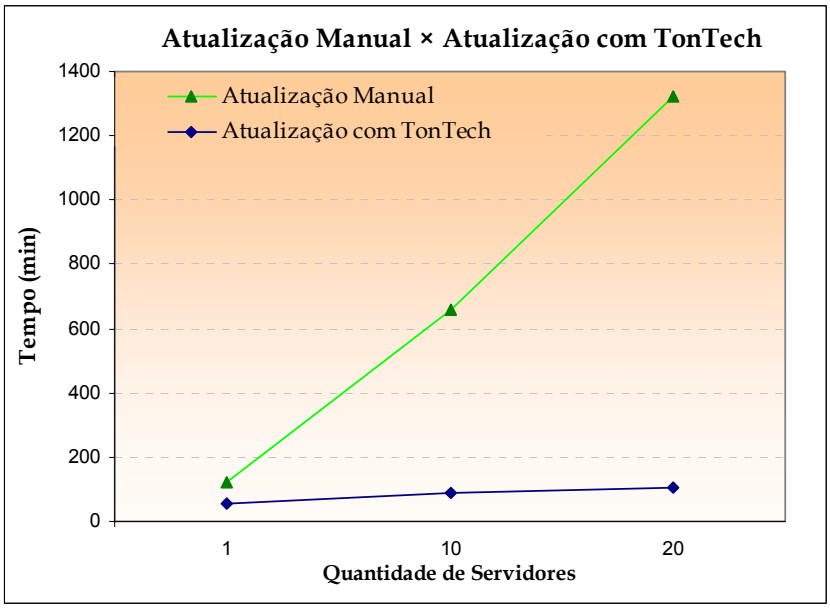

Figura 8. Comparação entre o tempo gasto para atualizar servidores manualmente × *TonTech*.

O tempo gasto para atualizar manualmente dez servidores não equivale proporcionalmente ao tempo utilizado para atualizar uma máquina. Isso ocorreu porque o *starter* da atualização foi feito em série nos servidores que foi efetuada paralelamente em todos eles.

No caso dos vinte servidores o tempo gasto coincidentemente dobrou em relação aos dez porque eles concorriam no acesso à internet - todos os casos de atualização manual.

6. CONSIDERAÇÕES FINAIS

A questão da necessidade de gerir os *patches* de segurança se mostrou um ponto de atenção, principalmente por empresas de TI com processos maduros, fato constatado durante as pesquisas sobre o tema. Um segmento que também valoriza tal prática são empresas de *outsourcing*, as quais buscam manter os ambientes dos clientes sempre de acordo com os níveis de atualizações recomendados pelos distribuidores de software. Entretanto, este cuidado implica em custos ao cliente.

Uma das grandes vantagens do *TonTech - Patch Management* é que empresas de pequeno/médio porte podem se valer da mesma filosofia das boas práticas, porém, de uma forma automatizada, segura e ausente de custos.

Pelos testes efetuados, pôde-se observar a economia de tempo ao se fazer a atualização dos pacotes do SO com o auxílio da ferramenta, além das outras funcionalidades proporcionadas aos responsáveis por esse tipo de função.

Esse trabalho mostra que toda solução para um determinado problema requer mais que apenas a implementação das idéias, necessitada estruturação da base de planejamento de projeto. Ou seja, ao exercitar os preceitos de gerência de projeto e engenharia de *software*, as etapas se relacionam como módulos para a construção do todo.

Como proposta de melhorias futuras e continuação do trabalho, indica-se a possibilidade da expansão da abrangência dos sistemas operacionais passíveis de serem administrados pela ferramenta. Além disso, o aperfeiçoamento do mecanismo de atualização dos pacotes.

REFERÊNCIAS

KROPIWIEC, Diogo Ditzel; GEUS, Paulo Lício de. **Paradigmas de segurança em sistemas operacionais**. 2004. Disponível em: <http://www.las.ic.unicamp.br/paulo/papers/2004-WSeg-diogo.kropiwiec-sistemas.operacionais.pdf>. Acesso em: 13 ago. 2007.

MACEDO, Rodrigo. **Quando o gerenciamento de patches se torna mais importante que remendar**, 2006. Disponível em: <http://www.microsoft.com/brasil/technet/colunas/rodrigomacedo/gerpatches.mspx>. Acesso em: 13 ago. 2007.

MATTOS, Mauro Marcelo. **Problemas de segurança em sistemas operacionais**: uma questão em aberto há quase 30 anos, 2004. Disponível em: <http://www.inf.furb.br/seminco/2004/artigos/124-vf.pdf>. Acesso em: 13 ago. 2007.

TANENBAUM, Andrew S.; FURMANKIEWICZ, Edson. **Sistemas operacionais**: projeto e implementação. 2. ed. Porto Alegre: Bookman, 2006.