# Projeto MAAOS – Mecanismo para monitoramento de Sistemas Operacionais e auditoria para detecção de vestígios

Vitor Teixeira Costa, George Soares Fleury

*Abstract* - **A Model of Security, in a broader context, offers greater control over the assets of information and services available. [2] The mechanism, presented, makes a tangible assessment of the quality of services and accountability on the misuse or mismanagement of resources.**

**The Project MAAOS (Monitoring and Auditing of files and Operating Systems) is a system for monitoring the integrity of files and audit processes that periodically monitors one or more hosts and maintains detailed records of changes to the file system, users, groups, tasks and other processes residing in kernel modules. The hosts, previously established, are monitored regularly, according to resources required and, if desired, the records can be kept for forensic purposes, while the administrator aware of possible attacks and / or threats, to isolate changes that indicate a change in pattern of a process.**

**After the adoption and implementation of this project, the protection of corporate data on the integrity, confidentiality and availability, in terms of operating systems, will cover an automated system, ensuring greater flexibility in the identification of security incidents that affect the corporate services in accordance due to its degree of criticality.**

*Index Terms* — **Monitoring. Audit. Integrity. Confidentiality. Availability.**

## I. Introdução

A criação de ferramentas automatizadas contendo informações pertinentes, além de base normativa, é uma necessidade quando se tem um grande volume de servidores. [1] Executar diariamente procedimentos operacionais (monitoramento, auditoria, gestão de usuários, contas de serviço) em muitos servidores, pode levar a falhas no próprio procedimento bem como ser exaustivo ao operador ao longo do tempo, fazendo com que esse possa ter a possibilidade de mascarar os resultados, mesmo que involuntariamente.

A intervenção humana não é aceita em processos de auditoria ou verificação de estado de sistemas. [1] Nesse princípio, a criação de procedimentos padronizados automáticos se faz necessária, de forma a conter os danos causados por possíveis ataques e resguardar os ativos de informação contra roubo, vandalismo, negligência ou imperícia e catástrofes, aderentes a procedimentos específicos.

Este projeto diferencia-se de outros softwares similares principalmente devido à funcionalidade de auditoria, uma vez que, se tratando de monitoramento de integridade de arquivos em sistemas operacionais, existem softwares já existentes no mercado e com nível de maturidade bastante elevado como o AIDE, Samhain e Tripwire. Estes, possuem diversas características semelhantes ao projeto sugerido, porém, a funcionalidade de auditoria, baseada na execução de *scripts*, está agregada somente ao MAAOS.

O objetivo principal deste projeto é sustentar a alta disponibilidade de serviços de rede e integridade de dados, de maneira a identificar com maior eficiência e agilidade, possíveis tentativas de ataque que comprometam a estrutura de um sistema operacional. Nesse contexto, podemos identificar os seguintes objetivos específicos do Projeto MAAOS:

- Criar mecanismos para monitoramento de integridade de servidores e auditoria de sistemas operacionais.
- Criar ambiente capaz de detectar automaticamente inconformidades nos servidores;
- Criar um ambiente portável para quaisquer plataformas de sistemas operacionais;
- Estabelecer pontos de controle e monitoramento do ambiente operacional por sensores de comportamento em um ambiente escalável;
- Emitir relatórios de auditoria e conformidade;
- Reduzir a intervenção humana nos processos de gestão em ambientes operacionais.

## II. Motivação

Os incidentes relacionados à segurança de dados em empresas seguem aumentando, proporcionalmente ao risco de ataques futuros. É o que se conclui do relatório estatístico emitido pelo Cert.br [9] acerca dos incidentes de segurança reportados no primeiro trimestre de 2009. Segundo a pesquisa, 0,53% dos ataques são do tipo web, um método de ataque visando especificamente o comprometimento de servidores Web ou desfigurações de páginas na Internet. Seguidos de 5,34% de *Scans*, notificações de varreduras em redes de

Manuscrito recebido em 21 de agosto de 2009.

Vitor Teixeira Costa é Analista de Segurança da Informação do Ministério das Relações Exteriores, Professor Universitário de cursos de Graduação e Pós-Graduação e Especialista em Perícia Digital e Redes de Computadores (e-mail: vitorteixeirac@gmail.com).

George Soares Fleury é Analista de Segurança da Informação do Ministério das Relações Exteriores, Analista de Suporte da Empresa Brasileira de Comunicações (EBC), Consultor oficial no Brasil do projeto Openbsd.org, com a empresa OpenSolutions (2001 até 2006) (e-mail: gfleury@gmail.com).

computadores, com o intuito de identificar quais computadores estão ativos e quais serviços estão sendo disponibilizados por eles. É amplamente utilizado por atacantes para identificar potenciais alvos, pois permite associar possíveis vulnerabilidades aos serviços habilitados em um computador. Paralelamente, 14,24% ataques de *Worm*, notificações de atividades maliciosas relacionadas com o processo automatizado de propagação de códigos maliciosos na rede. E por fim 79,59% são tentativas de fraudes, segundo dicionário Houaiss, é "qualquer ato ardiloso, enganoso, de má-fé, com intuito de lesar ou ludibriar outrem, ou de não cumprir determinado dever; logro".

Um estudo conduzido pela divisão de segurança da Cisco [7] revela que, cada vez mais, as empresas brasileiras e seus clientes são alvos de ataques na internet. Para se ter uma ideia do problema, entre 23 de abril e 6 de maio, sete diferentes tipos de vírus e *malwares*, que tinham como alvo companhias do Brasil, foram identificados. Segundo o gerente de desenvolvimento de negócios da Cisco da área de segurança, o aumento das ameaças reflete uma maior organização dos grupos criminosos voltados para o roubo de informações. "Os ataques estão bem mais direcionados, com alvos bem definidos", explica. O Brasil, por ter uma economia crescente, se torna um foco interessante. Contribui para o aumento das ameaças, também, a falta de maturidade das empresas brasileiras nos investimentos em tecnologia. Segundo o executivo, não dá mais para pensar em segurança apenas pontualmente. As corporações devem encarar o problema de forma mais ampla, pensando no conceito de gestão de risco. Em contraste com a sofisticação tecnológica do País no setor financeiro, por exemplo, essa falta de maturidade na área de segurança gera um ambiente atrativo para os criminosos.

Juntas, as previsões nos levam a um cenário de ameaças não necessariamente calmo, mas bem parecido com o de 2008, exceto pelo fato de que tudo vai se multiplicar. Para a Websense, por exemplo, mais de 80% do conteúdo malicioso estará hospedado em sites com boa reputação em 2009. Aqueles que realizam os ataques terão um modelo de distribuição de *botnets* e códigos maliciosos pela rede mais rápido e eficiente. Ficará mais difícil localizá-los e eliminá-los.

De acordo com a Cisco, o grande perigo são os criminosos virtuais, que procuram aumentar suas chances de sucesso por meio de técnicas mistas de ataques, combinando e-mail, web e sistemas intrusos. Os *botnets* se tornarão mais versáteis e os criminosos vão usá-los para hospedar *malwares* ou atacar alvos específicos. Com o aumento de trabalhadores remotos, do uso de ferramentas baseadas na web, de equipamentos portáteis e de tecnologias de virtualização, a preocupação deve aumentar. Para a Verisign, o foco dos ataques serão os sistemas SCADA, que realizam controle crítico de infra-estruturas. A crise financeira global, as fusões empresarias, os processos de consolidação e os colapsos de sistemas gerarão "oportunidades sem precedentes" para os criminosos virtuais, que sempre estarão atentos a esses movimentos para realizar ataques. De acordo com a Websense, soluções RIA (*rich internet application*) e ambientes de *cloud computing* são alvos certos. Na medida em que a popularidade das RIAs crescem, há também um crescimento de ataques que tiram vantagem das vulnerabilidades encontradas em seus componentes núcleo para que criminosos tomem controle remoto do sistema do usuário. Os ambientes de *cloud computing*, por sua vez, podem ser utilizados para o envio de *spam* e hospedagem de códigos maliciosos.

Todos esses dados, nos levam a crer da necessidade cada vez mais evidente da monitoração efetiva dos principais recursos presentes na infra-estrutura das corporações.

### *A. Segurança em Sistemas Operacionais*

A segurança do servidor é tão importante quanto a segurança da rede, pois frequentemente armazenam grande parte das informações vitais de uma empresa. Se um servidor for comprometido, todo o seu conteúdo pode ficar disponível para o *cracker* roubar ou manipular como quiser. [8] Abaixo, são descritos alguns exemplos de possíveis ameaças aos sistemas operacionais.

**Serviços não utilizados e portas abertas:** Uma ocorrência comum dentre administradores de sistemas é instalar o sistema operacional sem prestar atenção em quais pacotes realmente estão sendo instalados. Isto pode ser problemático, pois serviços desnecessários podem ser instalados, configurados com opções padrões e possivelmente acionados. Isto pode fazer com que serviços muitas vezes não necessários, como Telnet, DHCP ou DNS, sejam executados em um servidor ou estação de trabalho sem que o administrador perceba, o que pode causar tráfego indesejado no servidor, ou até mesmo uma via potencial para *crackers* ao sistema.

**Serviços não configurados:** A maioria das aplicações de servidor inclusas em uma instalação padrão são sólidas. Sendo utilizadas em ambientes de produção por muitos anos, seus códigos têm sido constantemente refinados e muitos dos erros (*bugs*) são encontrados e consertados. Entretanto, não existe software perfeito e sempre há espaço para mais aprimoramento. Além disso, softwares mais novos frequentemente não são rigorosamente testados como se espera, porque chegam recentemente a ambientes de produção ou porque talvez não sejam tão populares quanto outros software de servidor. Desenvolvedores e administradores de sistemas frequentemente encontram erros exploráveis em aplicações de servidor e publicam a informação em sites de rastreamento de erros e relacionados a segurança, como a lista de discussão Bugtraq (http://www.securityfocus.com) ou o site do Computer Emergency Response Team (CERT) (http://www.cert.org). Apesar destes mecanismos serem maneiras efetivas de alertar a comunidade sobre vulnerabilidades de segurança, é responsabilidade dos administradores consertarem seus sistemas prontamente. Isto requer uma atenção especial, pois os *crackers* têm acesso aos mesmos serviços de rastreamento de vulnerabilidades e utilizarão as informações para violar sistemas não corrigidos sempre que puderem. Uma boa administração de sistemas requer vigilância, constante rastreamento de erros e

manutenção apropriada do sistema para assegurar um ambiente computacional mais seguro.

**Administração desatenta**: Administradores que não configuram seus sistemas apropriadamente são grandes ameaças à segurança de servidores. De acordo com o System Administration Network and Security Institute (SANS), a causa fundamental da vulnerabilidade na segurança de computadores é "delegar pessoas não treinadas para manter a segurança e não prover nem treinamento nem tempo para que o trabalho seja executado". Isto se aplica tanto para administradores inexperientes quanto para administradores super-confiantes ou desmotivados. Alguns administradores falham em configurar seus servidores e estações de trabalho, enquanto outros falham em monitorar as mensagens de registro do *kernel* do sistema ou tráfego de rede. Outro erro comum é deixar senhas ou chaves padrões de serviços inalteradas. Por exemplo: alguns bancos de dados têm senhas de administração padrão porque seus desenvolvedores assumem que o administrador de sistemas irá alterá-las imediatamente após a instalação. Se um administrador de banco de dados deixar de alterar esta senha, mesmo um *cracker* inexperiente pode utilizar uma senha padrão conhecida para obter privilégios administrativos ao banco de dados. Estes são apenas alguns dos exemplos de como uma administração desatenta pode desencadear no comprometimento de servidores.

**Serviços essencialmente inseguros:** Até a empresa mais atenta pode ser vítima de vulnerabilidades se os serviços de rede forem essencialmente inseguros. Por exemplo, há muitos serviços desenvolvidos sob a suposição de que são utilizados através de redes confiáveis, mas essa suposição deixa de ser verdadeira a partir do momento em que o serviço for disponibilizado através da internet, que por si só é essencialmente não confiável. Um tipo de serviço de rede inseguro são aqueles que requerem nomes de usuário e senha não criptografados para autenticação. Telnet e FTP são dois serviços deste tipo. Se um software de *sniffer* de pacotes está monitorando o tráfego entre o usuário remoto e um servidor deste tipo, os nomes de usuário e senhas podem ser interceptados facilmente. Tais serviços também podem ser facilmente enquadrados no que a indústria da segurança chama de ataque *man-in-the-middle*. Neste tipo de ataque um *cracker* redireciona o tráfego de rede, enganando um servidor de nomes da rede, apontando-o para sua máquina ao invés do servidor pretendido. Quando alguém abrir uma sessão remota para este servidor, a máquina do atacante age como um condutor invisível, situado discretamente entre o serviço remoto e o usuário, capturando as informações. Desta maneira um *cracker* consegue obter senhas administrativas e dados sem que o servidor ou o usuário perceba.

Outra categoria de serviços inseguros são sistemas de arquivos de rede e serviços de informação, como NFS ou NIS, desenvolvidos explicitamente para uso em LANs, mas que, infelizmente, têm seu uso estendido para WANs (para usuários remotos). O NFS não tem, por padrão, nenhuma autenticação ou mecanismos de segurança configurados para prevenir um *cracker* de montar o compartilhamento do NFS e acessar qualquer coisa contida nele. O NIS também tem informações vitais que devem ser conhecidas por qualquer computador ou rede, incluindo senhas e permissões de arquivo, em um banco de dados ASCII ou DBM (derivado do ASCII) somente-texto. Um *cracker* que obtém acesso a este banco de dados poderá acessar qualquer conta de usuário de uma rede, inclusive a conta do administrador.

Diante do crescimento da complexidade das invasões e dos altos índices de tentativas bem sucedidas, observamos que os falsos negativos são problemas reais em nossa infra-estrutura e impossíveis de ser mensurados. Ocorrem quando os equipamentos de segurança não indicam que um ataque está sendo efetivado. Os falsos negativos são mais perigosos do que os falsos positivos, apesar de ambos serem indesejáveis, pois atribui ao Administrador a sensação de estar com o ambiente seguro e controlado.

Portanto, não há solução única para resolver todos os problemas, a segurança deve ser observada em um contexto mais amplo, capaz de mitigar todas as possibilidades de ataque e, caso ocorram, prover dados para contramedidas. A solução definitiva não será a implantação de apenas um recurso, mas sim, a combinação de diversos mecanismos, objetivando cercar todas as possibilidades de falsos negativos. Assim, o MAAOS será responsável pelo monitoramento e auditoria do sistema operacional de todos os servidores considerados críticos do ambiente.

## III. Ambiente Operacional

O MAAOS é um sistema de monitoramento de integridade de arquivos e auditoria de processos em sistemas operacionais, utilizado para acompanhar as criações, modificações ou exclusões de arquivos e relatar essas ações a uma console gerenciadora. O MAAOS recebe mensagens periódicas de um agente instalado no servidor e as armazena em um banco de dados MySql. Quando as mudanças são detectadas, o sistema irá registrar esses eventos no Banco de Dados e, opcionalmente, poderá enviar um e-mail para um usuário ou grupo de usuários.

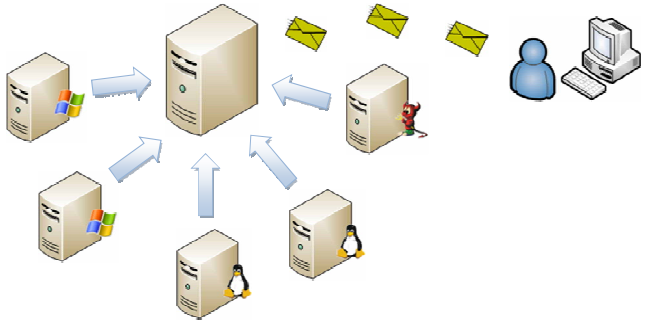

Figura 1 – Fluxo de comunicação

Outra importante funcionalidade é a capacidade de auditoria, no qual possibilita a verificação de *status* de quaisquer itens nos sistemas operacionais (usuários, grupos,

processos, serviços, partições, etc), desde que sejam criados *scripts* adequados ao processo e sistema operacional monitorado.

Todos os componentes do sistema MAAOS podem ser compilados e executados no Windows e em ambos os sistemas UNIX, incluindo BSD, Linux, Mac OS X e Darwin, AIX, IRIX. Isto permite uma grande flexibilidade para gerenciar todos os tipos de plataformas, quer a partir de um ambiente UNIX ou Windows.

Uma importante característica do sistema é o modo de atuação passivo, no qual apenas detecta e relata as mudanças ocorridas a console de gerenciamento. Essa estrutura se faz necessária para que falsos positivos não gerem ações que possam afetar a disponibilidade de sistemas íntegros que possuam similaridades com sistemas maliciosos e sejam detectados como ataques. Portanto, as contramedidas para um possível ataque ficarão a cargo do Administrador, que deverá determinar quais (se houver) ações deverão ser adotadas.

Como parte integrante do projeto, está sendo utilizado o software *open source* Osíris [6], em que foram realizadas várias modificações e inclusões de funcionalidades. Dentre as principais, podemos citar modificações realizadas nos agentes para agregar a funcionalidade de auditoria, utilização do Banco de Dados MySQL, interface gráfica para administração e todo o mecanismo para possibilitar a execução de *scripts* remotos, funcionalidades estas não presentes no Osíris.

### A. *Monitoramento*

Monitoramento é a observação e o registro regular das atividades de um determinado processo. É um procedimento rotineiro de acúmulo de informações em todos os seus aspectos, sendo um componente do ciclo de gestão que permite verificar a efetividade e eficiência do processo de execução ou ação, a fim de identificar suas realizações e fraquezas e recomendar medidas corretivas para otimizar os resultados desejados [1].

As informações que precisam ser coletadas no processo de monitoramento dependem, evidentemente, dos objetivos e ações do projeto ou programa e das forças e fatores externos que os influenciam.

No projeto MAAOS, o monitoramento será responsável por checar quaisquer modificações nos arquivos previamente definidos do sistema operacional monitorado, ou seja, uma observação sistemática e com propósitos. Num primeiro momento, o Administrador deverá criar um servidor íntegro, capaz de prover uma base de informações para as comparações futuras. Esta funcionalidade, após a comparação, deverá retornar o *status* dos arquivos monitorados e algumas informações importantes dos arquivos que possivelmente sofreram algum tipo de alteração.

### B. *Auditoria*

A auditoria é uma atividade que engloba o exame de operações, processos, sistemas e responsabilidades gerenciais de uma determinada entidade, com intuito de verificar sua conformidade com certos objetivos, políticas institucionais, regras, normas ou padrões [3].

Neste contexto, a auditoria torna-se importante devido às seguintes considerações:

- Detectar desvios e variações com relação às normas estabelecidas pela política de segurança;
- Descobrir pontos vulneráveis ou problemas na concretização da política de segurança adotada;
- Prevenir possíveis incidentes e/ou problemas causados pela inconformidade com as políticas de segurança;
- Avaliar causas de incidentes e/ou problemas detectados nos sistemas auditados.

Para a realização de uma auditoria que contribua efetivamente para a alta disponibilidade dos recursos e seja concisa numa possível análise forense para detecção de eventual ataque, deverão ser levantados os principais processos que serão auditados frequentemente. Essa definição será a base para a criação de *scripts* apropriados a real necessidade de auditoria.

Na tabela abaixo, estão descritos alguns exemplos de itens que poderão ser verificados durante a auditoria.

| Windows | Linux |
|---|---|
| Tarefas agendadas | Tarefas agendadas |
| Conexões TCP e UDP abertas | Conexões TCP e UDP abertas |
| Usuários com sessão ativa | Usuários com sessão ativa |

Tabela 1 – Exemplos de itens de auditoria

### C. *Componentes*

O Projeto MAAOS é constituído por quatro componentes principais: um agente, a console de gerenciamento, um aplicativo de gerenciamento e uma interface gráfica. Para garantir a segurança das informações, faz uso de OpenSSL para a criptografia e autenticação em todos os componentes. Entre esses componentes, a comunicação ocorre no seguinte sentido:

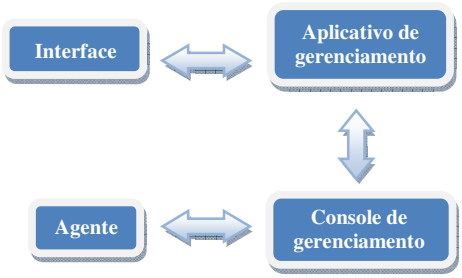


Figura 2 – Fluxo de comunicação dos elementos internos do sistema

*Agente*

Uma varredura do agente é um processo que ocorrerá de acordo com a periodicidade estabelecida em cada *host* monitorado (por padrão diariamente). O agente é responsável por varrer todos os locais previamente estabelecidos e gerar um *hash* MD5 ou SHA de cada arquivo para comparação. Em seguida, os dados são enviados a console de gerenciamento. Durante a varredura, há possibilidade de inúmeras verificações para controle e auditoria do sistema operacional, bastando

apenas à criação e inclusão no Banco de Dados de *scripts* de acordo com o sistema operacional. Essa funcionalidade deve ser atribuída ao arquivo de configuração específico do servidor a ser auditado, gerando a possibilidade de considerar as particularidades de cada servidor. Todas as informações coletadas são armazenados no Banco de Dados MySql da console de gerenciamento.

*Console de gerenciamento*

Realizar uma interação com confidencialidade e autenticidade dos dados que trafegam entre o agente e a console de gerenciamento é de imprescindível importância, pois é onde todas as informações sobre os servidores são mantidas incluindo configurações, *scripts* de auditoria, registros e as bases de dados. Por isso, no momento da inclusão de determinado servidor para início do monitoramento, são negociadas chaves criptográficas assiméticas. Nesse componente, também ocorrem às comparações dos *hashes* gerados pelos agentes com uma base de *hashes* íntegra gerada do mesmo servidor.

Em sistemas UNIX, assim como o Linux, o agente é executado com privilégio de separação. O processo só existe com privilégio de root para abrir arquivos durante uma varredura.

*Aplicativo de gerenciamento*

O aplicativo de gerenciamento é utilizado para verificação do monitoramento e administração dos servidores que participarão do processo. Comunica-se diretamente com a console de gerenciamento para obtenção dos dados e fornece esses dados para a Interface Gráfica.

Todas as funcionalidades para administração dos agentes instalados nos servidores monitorados e recuperação de informações, são realizadas através do aplicativo de gerenciamento.

*Interface*

A interface gráfica é constituída de um sistema web que se conecta a um Banco de Dados MySql, onde detém todos os dados dos servidores enviados pelos agentes. Trata-se de uma interface simples e eficaz. O exemplo da figura abaixo demonstra na tela principal, todos os servidores monitorados.


Figura 3 – Tela dos servidores monitorados

Foram criadas várias funcionalidades para facilitar a administração dos recursos disponibilizados pelo MAAOS. Destacamos as funcionalidades de monitoramento e auditoria, onde estão efetivamente as informações críticas.

Na funcionalidade de monitoramento, caso algum arquivo/diretório monitorado tenha sido modificado, criado ou excluído, seja em nível de conteúdo do arquivo ou até mesmo no tempo de acesso, um registro dessa ação será enviada ao Banco de Dados e aparecerá na interface. Na figura 4, é demonstrada a criação de novas portas no servidor monitorado. Com um clique sobre a linha que contém o registro, poderá ser verificado um maior nível de detalhamento. Para que este registro seja retirado da tela de monitoramento, o administrador deverá documentar a ciência do fato e a eventual contramedida realizada. Outras funcionalidades de monitoramento também podem ser encontradas nesta tela, como o status do agente em cada servidor e um mecanismo de busca, para identificação de registros antigos.

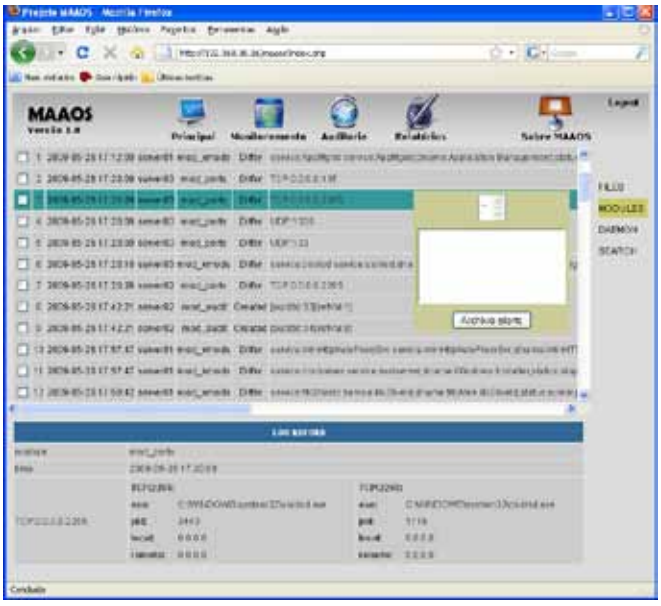


Figura 4 – Novas portas abertas, identificadas após a varredura do agente

## IV. RESULTADOS

Diante dos conceitos abordados, temos alguns resultados previstos que possivelmente ocorrerão após a implementação e outros resultados imprevisíveis, que dependerão da infra-estrutura onde se está implementando e principalmente dos objetivos traçados para auditoria.

Com intuito de demonstrar a eficiência do MAAOS, foi criado um laboratório com vários servidores, utilizando os principais sistemas operacionais utilizados atualmente para serviços críticos em infra-estruturas corporativas. Foram instalados dois servidores com o sistema operacional Windows Server 2003 Enterprise, um servidor com Windows Server 2008 Enterprise, dois servidores com o Linux Debian 2.6 e um servidor com o FreeBSD 7.2. O ambiente descrito foi colocado em produção durante uma semana para a realização de monitoramento e auditoria em plataformas distintas. A

tabela abaixo demonstra o nome dos servidores e o sistema operacional instalado.

| Nome do servidor | Sistema Operacional |
|---|---|
| server01 | Windows Server 2008 Enterprise |
| server02 | Windows Server 2003 Enterprise |
| server03 | Windows Server 2003 Enterprise |
| server04 | Linux Debian 2.6 |
| server05 | Linux Debian 2.6 |
| server06 | FreeBSD 7.2 |

Tabela 2 – Sistemas operacionais dos servidores monitorados

### A. *Monitoramento*

O produto do monitoramento são logs de criações, modificações ou exclusões realizados nos arquivos ou diretórios monitorados. Conforme demonstrado na figura abaixo, está sendo monitorado o diretório "maaos", localizado na raiz dos servidores. Durante o período de testes, foram criados vários arquivos, dentre eles "ameaça.dll", demonstrado na linha selecionada.

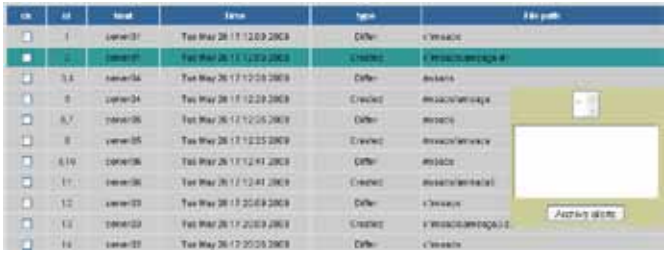

Figura 5 – Arquivos monitorados criados/modificados

Outras importantes informações são as datas relacionadas ao arquivo monitorado. A figura 6 especifica o detalhamento de um arquivo modificado. Na console houve registro do valor "Differ" no campo "type", que consiste no status de um arquivo modificado. O campo "file" refere-se ao nome desse arquivo. Em "time" identificamos o horário que as informações foram registradas no sistema. O valor de "bytes" informa o tamanho do arquivo inicial e o tamanho do arquivo após a modificação. No exemplo, de 0 bytes passou para 2 bytes. Por fim, o campo "mtime" registra o horário que o arquivo foi criado e o horário da última modificação.

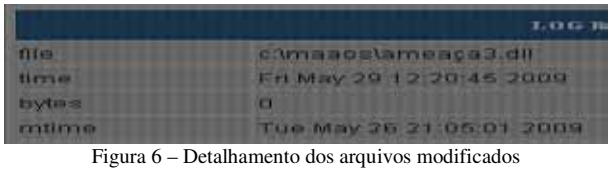


Figura 6 – Detalhamento dos arquivos modificados

### B. *Auditoria*

O produto da auditoria dependerá do *script* associado ao servidor. Esta funcionalidade proporciona uma grande variedade de possibilidades de auditoria no servidor. Considerando estruturas de sistemas operacionais totalmente distintos, poderão ser adicionados quaisquer *scripts* ou arquivos executáveis com os principais tipos de extensões de cada sistema operacional. Por exemplo, podemos associar um executável (.exe) para verificar os processos em execução numa máquina Windows e associar um outro *script* (.sh) para realizar a mesma ação numa máquina Linux.

No laboratório criado, foram associados alguns *scripts* para cada tipo de sistema operacional, descrito na tabela abaixo.

| Nome do servidor | Scripts de auditoria associados |
|---|---|
| server01 | Tarefas agendadas |
| server02 | Tarefas agendadas e conexões TCP e UDP abertas |
| server03 | Tarefas agendadas e conexões TCP e UDP abertas |
| server04 | Usuários com sessão ativa e tarefas agendadas |
| server05 | Usuários com sessão ativa e conexões TCP e UDP abertas |
| server06 | Usuários com sessão ativa |

Tabela 3 – *Scripts* de auditoria associados aos servidores

O resultado da auditoria no server04, demonstrado na figura 7, evidencia a saída dos *scripts* associados ao servidor. Nessa tela temos o nome do *script*, o horário de execução e as informações coletadas. Na primeira linha de dados, o *script* linux_users.sh, retornou os usuários com sessões ativas no servidor, explicitando inclusive o endereço lógico de uma conexão remota (192.168.33.62). O *script* linux_sessoes.sh, demonstra todas as conexões TCP e UDP abertas com computadores externos, os endereços lógicos (IP), as portas utilizadas e o *status* da conexão.

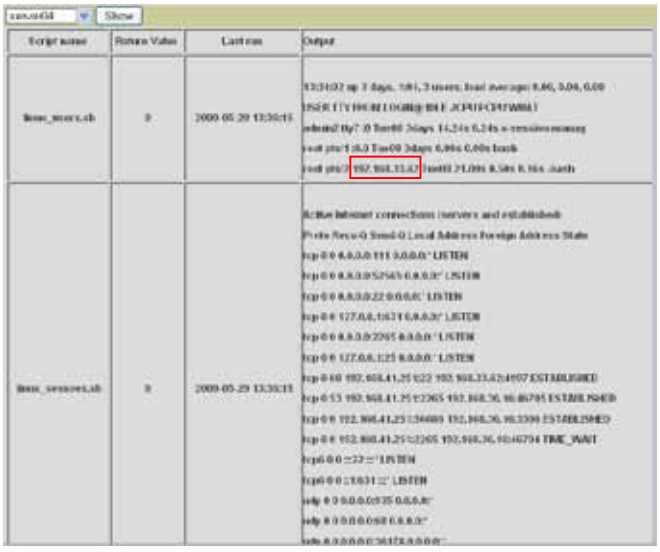

Figura 7 – Execução de *script* no servidor server04

## V. Trabalhos Futuros

- Aprimoramento da interface gráfica
- Desenvolvimento de interface para o aplicativo de gerenciamento
- Criação de *scripts* de auditoria para os diversos tipos de sistemas operacionais
- Criação de relatórios gerenciais

## VI. Conclusões

Diante de novas demandas de segurança da informação em manter dados e sistemas com alto grau de disponibilidade, observamos a crescente necessidade de sistemas que controlem a integridade dos dados e possibilite a verificação de itens variáveis em sistemas operacionais de maneira automatizada.

Proteger a empresa contra a perda de dados é uma obrigação estratégica, com soluções de garantia de segurança abrangente que possibilitem que identidades confiáveis interajam com sistemas e acessem as informações de maneira livre e segura.[4] Estes acréscimos suportam a estratégia de gerenciamento de riscos à informação, que permite a equipe de infra-estrutura, eficazmente, reconhecer, avaliar e mitigar riscos à sua informação em qualquer ponto do seu ciclo de vida. É necessário proporcionar um meio de definir e executar padrões de segurança de configuração e conduzir testes de conformidade regulares automaticamente.[1] A automação bem-sucedida destas políticas, aliada ao gerenciamento padronizado de configurações de rede, garante que os dados estejam disponíveis e seguros, eliminando os riscos para o negócio. Isso resulta em uma redução da paralisação dos serviços e dos custos associados a processamento de *trouble ticket* e prejuízo envolvendo *Service Level Agreement* (SLA).

A solução apresentada retrata um mecanismo que fará parte da arquitetura de segurança da informação da corporação. Este deverá ser utilizado em situações bastante pontuais, no qual há necessidade de monitorações constantes de integridade de determinados arquivos e auditoria de itens relevantes nos sistemas operacionais.

Neste contexto, com intuito de preservar a estrutura de comunicação do sistema, foram inseridos aspectos importantes, como criptografia assimétrica entre cliente e servidor, e acesso web ao sistema via https, a fim de proporcionar um ambiente estável e contínuo, mitigando possíveis falhas para possibilitar a realização adequada dos objetivos. Assim, o Projeto MAAOS torna-se bastante eficiente na automatização de processos rotineiros de monitoração de itens dos sistemas operacionais que devem estar de acordo com a política de segurança da corporação, necessitando da intervenção humana apenas para situações de contramedidas.

## Referências


[1] WOTRING, Brian. *Host Integrity Monitoring Using Osiris and Samhain*. Editora: Syngress.

[2] KUROSE, James F. e ROSS, Keith. *Redes de Computadores e a Internet*: Uma nova Abordagem. Editora: Addison Wesley, 2006.

[3] DIAS, Cláudia. *Segurança e Auditoria da Tecnologia da Informação.* Editora: Axcel, 2000.

[4] WOTRING, Brian. *Host Integrity Monitoring: Best Practices for Deployment.* (2009, Março 02). [Online]. Disponível: http://www.securityfocus.com/infocus/1771

[5] DUNSTON, Duane. *Mass deploying Osiris* (2009, Março 13). [Online]. Disponível: http://www.linuxsecurity.com/content/view/101884/49/

[6] *Osíris – Documentação Oficial* (2009, Abril 20) Disponível: http://osiris.shmoo.com

[7] Cisco Systems. *Documentação Oficial* (2009, Junho 10) Disponível: http://www.cisco.com/

[8] Red Hat. *Documentação oficial* (2009, Junho 20). [Online]. Disponível: http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-sg-pt_br-4/s1-risk-serv.html

[9] Centro de Estudos Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.BR). *Dados Estatísticos.* (2009, Maio 26). [Online]. Disponível: http://www.cert.br/stats/incidentes/