JISTEM - Journal of Information Systems and Technology Management
Revista de Gestão da Tecnologia e Sistemas de Informação
Vol. 11, No. 3, Sept/Dec., 2014 pp. *717-734*
ISSN online: 1807-1775
DOI: 10.4301/S1807-17752014000300010

SEGURANÇA DA INFORMAÇÃO DE PRODUÇÃO E OPERAÇÕES: UM ESTUDO SOBRE TRILHAS DE AUDITORIA EM SISTEMAS DE BANCO DE DADOS

SECURITY INFORMATION IN PRODUCTION AND OPERATIONS: A STUDY ON AUDIT TRAILS IN DATABASE SYSTEMS

Rodrigo Roratto
Evandro Dotto Dias
Universidade Federal de Santa Maria, RS/Brazil

ABSTRACT

Special care should be taken to verify the integrity and to ensure that sensitive data is adequately protected. One of the key activities for data loss prevention is anaudit. And in order to be able to audit a system, it is important to have reliable records of its activities. Systems that store critical data, whether financial or productive, must have features such as audit log, also called audit trail, which records all activities on critical data. This allows to identify harmful actions that can be internal or external, intentionally or unintentionally caused. Therefore, this paper presents major studies in security audit trail (audit log), especially records of logs, and it presents what is available in terms of commercial tools and what they offer.
Keywords: Audit trails, Information Security, Computer systems management technologies available, Computational Risk.

RESUMO

Cuidados especiais devem ser aplicados para verificar a integridade e para garantir que dados sensíveis estejam adequadamente protegidos. Uma das atividades fundamentais para que se evitem estas perdas é a auditoria. E para ser possível auditar um sistema é importante que se tenha registros confiáveis das atividades no mesmo. Sistemas que armazenam dados críticos, sejam financeiros ou de ordem produtiva, devem dispor de recursos como o log de auditoria, também chamado de trilha de auditoria, que registre todas as atividades nos dados críticos. Isto permite identificar ações danosas que podem ser internas ou externas causadas intencionalmente ou não. Diante disso, neste estudo fez-se um levantamento das principais pesquisas na área de segurança de trilhas de auditoria (logs de auditoria), em especial registros de logs, e apresenta-se o que existe disponível em termo de ferramentas comerciais e o que elas oferecem.
Palavras-chave: Trilhas de auditoria, Segurança da informação, Sistemas informatizados de gestão, Tecnologias disponíveis, Risco computacional.

Manuscript first received/*Recebido em:* 23/09/2013 Manuscript accepted/*Aprovado em*: 17/09/2014

Address for correspondence / Endereço para correspondência

Rodrigo Roratto, Universidade Federal de Santa Maria, RS/Brasil. Email: roratto_rs@hotmail.com

Evandro Dotto Dias, Universidade Federal de Santa Maria, RS/Brasil, E-mail: evandrodotto@yahoo.com

1. INTRODUÇÃO

A informação é um recurso crítico nas organizações. Com a maior disponibilidade de computadores para usuários de todos os tipos em todo o mundo, mais dados estão sendo processados em curtos períodos de tempo. Entender o que implica a segurança do computador exige a compreensão dos termos exposição, vulnerabilidade e risco (Bosworth; Kabay, 2002).

A vulnerabilidade é uma fraqueza no sistema computacional ou em seus arredores que pode se tornar um risco à segurança.Um risco computacional é a probabilidade de um evento resultar em uma perda. Os riscos e as perdas podem incluir perdas financeiras e de pessoal, perda de reputação e base de clientes, incapacidade de funcionar de uma forma atempada e eficaz, a incapacidade de crescer, e violação das leis e regulamentações governamentais (Bosworth; Kabay, 2002). Nos sistemas em que o proprietário do dispositivo não é a mesma pessoa que é proprietária dos segredos dentro do dispositivo, é essencial que existam mecanismos de auditoria no local para determinar se houve alguma tentativa de fraude (Schneier; Kelsey, 1999).

Cuidados especiais devem ser aplicados para verificar a integridade e para garantir que dados sensíveis estejam adequadamente protegidos. Uma das atividades fundamentais para que se evitem estas perdas é a de auditoria. A tarefa de auditoria inclui recomendar ações para eliminar ou minimizar as perdas, através da identificação de vulnerabilidades e riscos, determinar se os controles de segurança adequados estão em vigor, garantir que os dispositivos de auditoria e segurança sejam válidos e verificar se os controles, trilhas de auditoria e medidas de segurança estão funcionando de forma eficaz (Bosworth; Kabay, 2002).

Em muitas aplicações o controle de acesso e outras informações relacionadas com as operações do usuário devem ser mantidos em arquivos de log seguros para detecção de intrusão e violações ou para fins de auditoria do sistema (Xu et al. 2005). Nos arquivos de log geralmente são armazenados uma grande quantidade de informações confidenciais. Portanto, é importante garantir que, caso ocorra alguma violação do sistema, os logs do mesmo não sejam comprometidos e a violação possa ser detectada posteriormente (Xu et al. 2005). O primeiro alvo de um atacante experiente será o sistema de log de auditoria: o atacante quer apagar os traços do ataque, para escapar da detecção, bem como manter o método de ataque em segredo para que as falhas de segurança explorada não sejam detectadas pelos administradores do sistema (Bellare; Yeey, 1997). Isto demostra que há uma grande nescessidade de se estudar e desenvolver técnicas que permitam gerir e principalmente garantir a inviolabilidade dos registros de auditoria para protegê-los de adulterações e danos causados por qualque tipo de usuário ou pessoa de forma intencional ou não. A partir destes conceitos buscar-se-á fazer neste estudo um levantamento dos principais estudos na área e ferramentas disponíveis no mercado.

Com base nestes problemas busca-se neste trabalho fazer um estudo e apresentar uma análise dos seus principais pontos que devem ser considerados na gestão de segurança da informação e uma descrição das principais tecnologias e pesquisas, referentes às formas de aplicação e proteção das trilhas de auditorias (logs). Também se faz uma breve descrição de duas das principais soluções comerciais disponíveis no mercado hoje. Com isto, busca-se encontrar soluções contra violações voluntárias ou involuntárias que podem partir de qualquer usuário ou até mesmo do administrador do sistema.

O presente trabalho está organizado da seguinte forma: primeiramente no item 2 é apresentada a metodologia utilizada neste estudo; no item 3, apresenta-se o conceito de segurança da informação. Posteriormente, no item 4, são apresentados os conceitos de trilhas de auditoria; a seguir, no item 5, descreve-se o uso de logs em sistemas gerenciadores de banco de dados. No item 6, são apresentados e descritos os principais trabalhos e pesquisas que buscam soluções para a proteção dos registros de auditoria; a seguir, no item 7 é apresentada uma breve descrição dos recursos disponíveis para a gestão de dados críticos oferecidos comercialmente pelas duas principais empresas fornecedoras de soluções para gestão e armazenamento de informações; por fim, no item 8, apresenta-se as conclusões deste estudo.

2. METODOLOGIA

Quanto aos procedimentos metodológicos, esta pesquisa caracteriza-se como um estudo descritivo bibliográfico focado em um estudo de caso, visto que descreve modelos de sistemas de auditoria e uma análise de atributos a serem considerados para a segurança da informação de sistemas de logs. Também busca identificar o que existe de mais significativo em pesquisas e soluções para o problema de segurança de trilhas de auditoria. O estudo limita-se à análise de referencial teórico referente a soluções para a proteção de trilhas de auditoria (logs) e das ferramentas disponíveis para a gestão de dados críticos no mercado. Salienta-se que o termo log de auditoria, nesse estudo, equivale à trilha de auditoria.

3. A SEGURANÇA DA INFORMAÇÃO EM SISTEMAS DE GESTÃO

É evidente que os negócios estão cada vez mais dependentes das tecnologias e estas precisam proporcionar confidencialidade, integridade e disponibilidade. Segundo Albuquerque (2002) e Krause (1999), há três princípios básicos para garantir a segurança da informação principalmente no que se refere a sistemas que envolvam questões financeiras, tais como:

• Confidencialidade: a informação somente pode ser acessada por pessoas explicitamente autorizadas. É a proteção de sistemas de informação para impedir que pessoas não autorizadas tenham acesso;

• Disponibilidade: a informação deve estar disponível no momento em que a mesma for necessária;

• Integridade: a informação deve ser recuperada em sua forma original (no momento em que foi armazenada). É a proteção dos dados ou informações contra modificações intencionais ou acidentais não-autorizadas.

Alguns autores defendem ainda que, para que uma informação seja considerada segura, o sistema que a administra ainda deve respeitar os seguintes critérios:

• Autenticidade: garante que a informação ou o usuário da mesma é autêntico;

• Não repúdio: não é possível negar (no sentido de dizer que não foi feito) uma operação ou serviço que modificou ou criou uma informação; não é possível negar o envio ou recepção de uma informação ou dado;

• Legalidade: garante a legalidade (jurídica) da informação; a aderência de um sistema à legislação; e as características das informações que possuem valor legal dentro de um processo de comunicação, onde todos os ativos estão de acordo com as cláusulas contratuais pactuadas ou a legislação nacional ou internacional vigente;

• Privacidade: foge do aspecto de confidencialidade, pois uma informação pode ser considerada confidencial, mas não privada. Uma informação privada deve poder ser vista / lida / alterada somente pelo seu dono. Garante ainda que a informação não será disponibilizada para outras pessoas (neste caso é atribuído o caráter de confidencialidade à informação). É a capacidade de um usuário realizar ações em um sistema sem que seja identificado.;
• Auditoria: rastreabilidade dos diversos passos de um negócio ou processo, identificando os participantes, os locais e horários de cada etapa. A auditoria aumenta a credibilidade da empresa e é responsável pela adequação da empresa às políticas legais e internas.
A todas estas ponderações acerca de critérios para a segurança da informação soma-se outra como estratégia de gestão da informação: a veracidade. Isto significa que a informação deve estar calcada em acontecimentos verídicos ou argumentos lógicos compatíveis com a necessidade da organização. Nesse sentido, não basta que a informação seja autêntica, pois sua fonte pode ser desonesta. Não basta a confiabilidade, mas também deve existir veracidade da informação.

4. TRILHAS DE AUDITORIA (LOGS)

A auditoria procura identificar e evitar ações suspeitas e fraudulentas por parte do usuário, coletando dados sobre suas atividades no banco de dados. As informações coletadas são analisadas a fim de descobrir problemas de segurança e sua origem (Simon et al. 2008). A principal funcionalidade de um serviço de auditoria é oferecer armazenamento seguro e permanente dos registros de log, para que eles possam detectar quando uma falha de segurança ocorreu (Xu et al. 2005).

A necessidade de identificar quais foram as ações e determinar os padrões suspeitos são importantes requisitos para a segurança do sistema. Além disso, a auditoria deve ser realizada de maneira independente e transparente, de forma que todas as informações relevantes devem ser catalogadas (Hawthorn et al. 2006).

Uma trilha de auditoria, que também pode ser chamada de log de auditoria, é usada para assegurar o fluxo preciso das transações em um sistema. Cada detalhe de uma fonte e entrada de um determinado documento ou transação deve ser feita com base em um relatório ou arquivo.

A auditoria digital permite a verificação do conteúdo de um sistema de arquivo em um determinado período no passado. A auditoria é um protocolo de contestação/resposta entre o auditor e o sistema de arquivo a ser auditado (Peterson et al, 2007).

A técnica de rastreamento pode ser aplicada em uma única transação para teste rápido. No entanto, para garantir que o controle funcione consistentemente, o teste deve cobrir grandes volumes de dados em diferentes períodos de tempo (Bosworth; Kabay, 2002). As trilhas de auditoria devem ser construídas como parte normal dos sistemas de controles internos. Alguns sistemas podem ser adquiridos com o recurso de registro de auditoria automatizados.

O log de sistema inclui uma entrada para cada operação aplicada ao banco de dados que possam ser necessárias para a recuperação de uma falha de operação ou falha do sistema (Elmasri; Navathe, 2004). Podemos expandir as entradas de log para que ele também inclua o número da conta do usuário e terminal on-line para que seja aplicado a cada operação registrada no log. Se qualquer adulteração com o banco de

dados é suspeita, uma auditoria do banco de dados é realizada, o que consiste em analisar o log para examinar todos os acessos e operações aplicadas ao banco de dados durante um determinado período de tempo (Elmasri; Navathe, 2004). Quando uma operação ilegal ou não autorizada é encontrada, o DBA pode determinar o número da conta usada para executar esta operação. Auditorias de banco de dados são particularmente importantes para bancos de dados sensíveis, que são atualizados por muitas transações e usuários, como um banco de dados bancário, que é atualizado por muitas caixas de banco (Elmasri; Navathe, 2004).

Para se preparar para uma auditoria futura, um sistema de arquivos gera os metadados de autenticação que consolidam (commit) no sistema de arquivos referente ao seu conteúdo atual. Esses metadados são publicados em um terceiro lugar. Para realizar uma auditoria, o auditor acessa os metadados e faz contestações ao sistema de arquivos e confronta as informações obtidas com as representadas nos metadados (Peterson et al. 2007).

O sistema deve estar preparado para resistir a ataques com a criação de históricos e versões falsas que passam pelo processo de auditoria (Peterson et al. 2007). Esta classe de ataque inclui a criação de versões falsas do arquivo de dados que coincidem com os metadados publicados, mas diferem dos dados utilizados na sua criação. Também inclui a criação de históricos falsos, inserção ou exclusão de versões em uma sequência sem detecção. Neste item descreveu-se a importância da auditoria como atividade que tem como objetivo garantir a segurança e continuidade do negócio.

5. LOGS EM SGBDS

Guardar logs em sistemas de arquivos não é recomendado, pois se algum arquivo for excluído, não haverá registro desta ação. Este é um problema que não ocorre em um banco de dados (Mcdowall, 2007).

A maioria dos SGBDs permite catalogar ações na base de dados, gerando logs de auditoria. Infelizmente, os métodos em geral não são transparentes e muitos requerem a criação de triggers para cada objeto analisado (Simon et al. 2008). O uso de triggers não é recomendado, pois onera o uso do banco de dados por adicionar rotinas que devem ser executadas a cada ação realizada (Sallachl, 1992).

A geração de dados de auditoria pode ser implementada através de funções genéricas ou através de políticas de uso do banco e logs automáticos.

Para que se construam os registros de log deve-se ter uma tabela separada para gravação de todas as entradas de trilhas de auditoria associadas à criação, modificação e exclusão de dados e registros na base de dados (Mcdowall, 2007). Independentemente da abordagem, cada pacote no sistema deverá ter uma trilha de auditoria individual. A vantagem dessa abordagem é que todas as entradas de auditoria estão intimamente associadas com os dados que representam. Esta abordagem permite que se pesquise de forma mais específica e rápida os registros de log (Mcdowall, 2007).

Como exemplo será apresentada a geração de logs do SGBD PostigreSQL. O PostgreSQL permite que no seu arquivo de configurações possa ser definido que o sistema gere registros sobre suas atividades. Dentre os parâmetros presentes neste arquivo está a criação de um log para cada atividade realizada pelo SGBD. Este log

pode conter, além de informações sobre acesso aos dados, diversas outras, como as mensagens de conexões, erros de autenticação e erros de consultas SQL.

Um exemplo de registros de log gerados pelo PostgreSQL é apresentado na Figura 1. Neste caso, o arquivo de configurações foi definido para gerar somente informações sobre acessos aos dados. Como podem ser observados, os registros são gerados a cada consulta realizada pelo cliente, nos quais estão presentes as informações sobre os tipos de consultas e duração das mesmas. São estes os parâmetros que serão utilizados como base para a realização da auditoria.

```
Org:127.0.0.1(57303);Tsmp:2007-06-24 17:04:02.291BRT;DB:db-curso;
Usr:us-curso;Cmd:SELECT;ID:467ece31.1601-5;LOG: duration: 9.861 ms
Org:127.0.0.1(57303);Tsmp:2007-06-24 17:04:02.669 BRT;DB:db-curso;
Usr:us-curso;Cmd:UPDATE;ID:467ece31.1601-6;LOG: duration: 376.788 ms
Org:127.0.0.1(57303);Tsmp:2007-06-24 17:04:02.697 BRT;DB:db-curso;
Usr:us-curso;Cmd:SELECT;ID:467ece31.1601-7;LOG: duration: 27.878 ms
```

Figura 1: Exemplo de log PostgreSQL (SIMON et al. 2008)**.**

6. ESTUDOS EXISTENTES PARA SEGURANÇA DE TRILHAS DE AUDITORIA

Nesta seção, apresentam-se trabalhos publicados que de forma direta ou indireta apresentam contribuições para o problema de violação dos registros de log.

6.1. Forward Integrity for Secure Audit Logs

Uma entrada de log é composta de uma data (tempo) e descrição do evento. Um atacante experiente tenta alterar ou destruir os dados de log correspondentes às suas tentativas de login atuais ou passadas (Bellare; Yeey, 1997). Os autores deste trabalho, Mihir Bellare e Bennet S. Yeey, introduzem uma nova propriedade de segurança chamado por eles de “integridade para a frente” (FI) baseada no modelo de geração de códigos de autenticação de mensagem (MACS).

O objetivo do FI é prevenir a alteração ou inserção de informações pelo atacante, mesmo quando os registros de log ficarem disponíveis para o atacante que obteve o controle de todo o sistema.

No sistema MAC se um invasor obtém a chave MAC ele poderá forjar todas as entradas do registro. No sistema FI a posse da chave em um determinado ponto no tempo não permite que o atacante forje as entradas do registro de uma data anterior a atual. Assim, o atacante não pode alterar o conteúdo do registo (Bellare; Yeey, 1997). Ele ainda pode apagar as entradas, mas os espaços serão visíveis no registro e, também, a transmissão ocasional do log para um sistema remoto atenua o efeito da exclusão dos registros.

6.1.1. Códigos de autenticação de mensagem (MACs)

Normalmente, os MACs são usados em um contexto de comunicação, onde o remetente e o destinatário compartilham uma chave secreta MAC. O remetente usa a chave para gerar MAC de uma mensagem e atribuí-lo à mensagem; o receptor, que conhece a chave MAC, pode restaurar o MAC e aceitar como verdadeiras apenas as mensagens para o qual o MAC regenerado corresponde ao MAC transmitido.

A segurança do modelo MAC está no fato de que é computacionalmente inviável para um adversário baseado em rede que não sabe a chave do MAC modificar as mensagens e os MACs para que o receptor aceite-as como verdadeiras. Uma vez que os logs de auditoria são simplesmente mensagens que são lidas e verificadas mais tarde por um destinatário e não (necessariamente) por uma rede, talvez se possa simplesmente anexar aos MACs as entradas de log de auditoria para protegê-los. A seguir apresenta-se um esquema de codificação de logs por MAC.

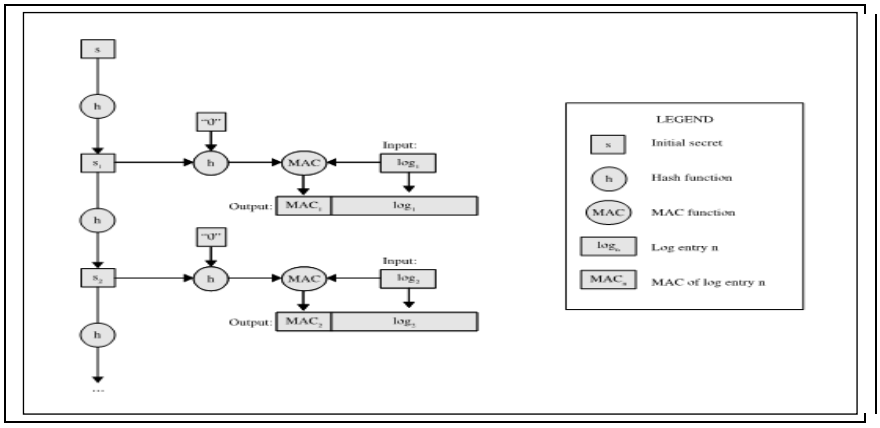

Figura 2: Exemplo de codificação por MAC (HOLT, 2006).

No entanto, o modelo MAC falha quando não é mantido um envio contínuo dos logs para um dispositivo remoto, seja por falta de um ou demora nas transferências. Outra vulnerabilidade é se um invasor entrar no sistema e obtiver a chave MAC ele obterá o controle do sistema de logs (Bellare; Yeey, 1997).

6.1.2. Modelo forward integrity (FI)

Este novo modelo proposto por Mihir Bellare e Bennet S. Yeey faz o uso de MACs de uma forma diferente, evitando a obrigatoriedade de replicação do log em registros remotos. Uma entrada de log consiste em uma data (tempo) e descrição do evento. Como citado anteriormente, um atacante experiente, que tenta comprometer os dados anteriores: ele deseja alterar, ou apagar, as entradas correspondentes às suas tentativas de login.

No modelo proposto mesmo que o atacante obtenha a chave em um dado período no tempo ele não conseguirá alterar os registros gerados anteriormente a este período (BELLARE; YEEY, 1997). Ele ainda pode apagar as entradas, mas os espaços serão visíveis no registro e, também, a transmissão ocasional do log para uma origem remota diminui as chances de destruição total dos registros.

Neste sistema, as chaves são mutáveis e evoluem em períodos de tempo, sendo geradas a partir da chave anterior. A chave Ki no período de tempo i é obtida com a função não reversível Ki-1 do intervalo de tempo anterior e o tempo atual. Após a nova chave Ki ser gerada, a chave Ki-1 é excluída. Sendo assim, se houver um ataque em i e o atacante obtiver a chave Ki, ele não conseguirá obter Kj para j<i. Isto evita que o atacante crie entradas de registro para períodos anteriores. Uma chave K0 é fornecida

para verificação da integridade de todas as versões independente do tempo. Neste artigo, é sugerido sistemas de log de auditoria onde:

$$log\_fn_j(m_i) = (m_i, FIMAC_j(m_i))$$

sendo que os geradores de mensagens log (mi) podem ou não ter controle sobre quando o registrador decide mudar os tempos. Para cada mensagem mi recebida em uma época Ej, o gerador de logs cria uma entrada log_fnj (mi) e armazena no log de auditoria, que pode ser posteriormente verificada. A FIMACj (mi) é um código de autenticação anexada às mensagens.

A proteção criptográfica deve ser feita com muito cuidado. Não basta criptografar com uma chave pública e fazer verificações. Se o atacante obtiver a chave de criptografia, ele simplesmente irá apagar o registro original e gerar os seus próprios logs. Além de criptografar esses segredos, o gerador de logs deve autenticar a sua utilização no início do log, talvez usando um protocolo timestamping (BELLARE; YEEY, 1997). Isto exigiria algumas comunicações de rede ou um esquema de assinatura digital baseado em chave. Ao invés de gerar os códigos no logger, a chave pode ser gerada por um dispositivo externo. Por exemplo, o log é gerado de forma segura e entregue ao logger usando um protocolo de criptografia com privacidade para a frente (FI). Manter os registros no arquivamento também é fundamental para não permitir que um invasor recupere os logs de um período anterior, algo que só poderá ser evitado se for utilizada uma chave de verificação do sistema (FI).

6.2. **Building an Encrypted and Searchable Audit Log**

Neste trabalho os autores desenvolvem um estudo para criar um mecanismo de busca de palavras-chave e criptografia dos arquivos de log. Delegação de recursos é importante para que um investigador possa pesquisar e encontrar entradas específicas em arquivos de log (WATERS et al. 2004). Eles desenvolveram um sistema com base em chaves que permite pesquisar palavras-chave em dados criptografados, utilizando o *Identity-Based Encryption* (IBE).

Se em algum momento o auditor quiser pesquisar um log de auditoria para identificar as entradas que combinam com uma determinada palavra-chave, ele deverá ir ao agente depositário de auditoria. Se o agente depositário considerar adequado, ele concede essa capacidade para o auditor. Ele, então, pode pesquisar através das entradas e ver que as entradas correspondem à palavra-chave. Para as entradas de log de auditoria que correspondem à palavra-chave, o investigador pode descriptografar a entrada e exibir seu conteúdo, conforme figura 4.

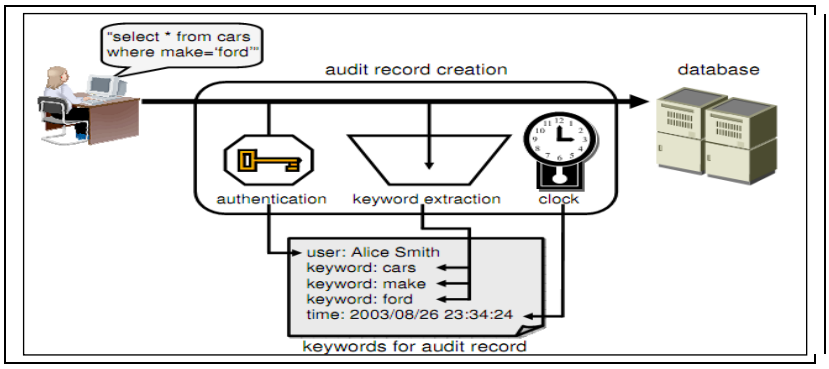

Figura 3: Esquema de busca em registros de log criptografados (WATERS et al. 2004).

6.2.1. **Funcionamento do Sistema**

As operações no regime assimétrico são significativamente mais caras do que as do regime simétrico. Os principais gargalos são os cálculos de exponenciação modular de emparelhamento para cada palavra-chave (Waters et al. 2004). No entanto, se as mesmas palavras-chaves são usadas com frequência, os resultados intermediários podem ser reutilizados. O modelo assimétrico corrige muitos dos inconvenientes do regime simétrico (Waters et al. 2004). Uma vez que cada servidor armazena apenas parâmetros públicos, não existem chaves secretas para um invasor roubar. Este modelo não permite ao atacante pesquisar ou decifrar nenhuma entrada no log de auditoria que já foram geradas e armazenadas.

No referido trabalho foi implementado um sistema de banco de dados de log de auditoria que cria entradas assimetricamente criptografadas e pesquisáveis. O agente de log é implementado como um servidor proxy MySQL. O servidor proxy, ao receber uma consulta, registra a consulta, além de passá-la para o servidor de banco de dados MySQL.

O proxy foi desenvolvido em uma plataforma Linux e é multi-threaded para que vários usuários possam ser atendidos simultaneamente em que o componente de log é executado em paralelo com o resto do sistema. O servidor de log de auditoria atribui a data e a hora para a cada entrada de log de auditoria.

As entradas de log são escritas em outro servidor de banco de dados MySQL, que é dedicado para armazenar as entradas de log. O software tem um servidor de cache que é usado para reutilizar buscas. Ele é implementado como uma tabela hash simples que associa o resultado com a palavra buscada.

Também foi implementado o método check pointing da cadeia hash. O servidor de log de auditoria calcula o valor atualizado da cadeia de hash para cada entrada de log que constrói. O valor de hash atual pode ser lido a qualquer momento. Uma parte que lê este valor pode usá-la mais tarde para verificar a integridade do log.

Nesta pesquisa os autores apresentaram a implementação de um modelo que permite fazer buscas através de palavras-chaves nos registro de log e garante a proteção destes registros através do modelo de chaves assimétricas.

7. SOLUÇÕES PARA SEGURANÇA DE LOGS EM SGBDS

Segurança e funcionalidade é o maior problema em qualquer SGBD (Jangra et al. 2010). Existem muitos no mercado de DBMS. Atualmente, temos diversos bancos de dados onde, como exemplo, pode-se citar: Alpha Five, DataEase, Oracle database, IBM DB2, Adaptive Server Enterprise, FileMaker, Firebird, Ingres, Informix, Mark Logic, Microsoft Access, Microsoft SQL Server, Microsoft VisualFoxPro, MonetDB, MySQL, PostgreSQL, Progress, SQLite, Teradata, CSQL, OpenLink Virtuoso, Daffodil

DB, OpenOffice.org Base etc.

Dentre estes bancos apresentados, os que são mais presentes no mercado são: Oracle, IBM e DB2 (JANGRA et al. 2010). A seguir são apresentados os recursos que cada um oferece para a gestão dos registros de auditoria.

7.1. Oracle® Database Vault

Oracle® Database Vault, na versão 11g, ajuda a proteger dados confidenciais de aplicativos até mesmo do acesso dos usuários privilegiados. Dessa forma, os clientes podem aumentar a proteção de seus dados confidenciais em aplicativos do acesso não autorizado de qualquer usuário, inclusive os altamente privilegiados, incluindo DBAs de aplicações potentes e outros, de acessar a dados e aplicações sensíveis nas bases de dados Oracle fora do âmbito das suas responsabilidades (ORACLE, 2010). Por exemplo, você pode restringir o acesso administrativo para salários de funcionários, registros de clientes médicos, ou outras informações confidenciais.

Pode ser usado ainda para determinar a separação de obrigações dentro do banco de dados, por exemplo, bloqueando o acesso do DBA a dados confidenciais dos aplicativos, porém permite que ele execute as operações do dia a dia como backup e recuperação, ajuste e replicação do banco de dados. Permite consolidar os bancos de dados dos aplicativos e determinar fronteiras e políticas sólidas em torno do acesso a esses dados.

Regulamentações como Sarbanes-Oxley (SOX), Healthcare Insurance Portability and Accountability Act (HIPAA), Basiléia II e Data Security Standards (DSS) da Payment Card Industry (PCI) exigem que as empresas considerem a separação de obrigações e controles rigorosos de acesso para as informações confidenciais conforme tabela 1 (ORACLE, 2010).

Tabela 1: Os regulamentos que ameaças potenciais à segurança (ORACLE, 2010).

Regulamento	**Ameaça potencial à segurança**
Sarbanes-Oxley Section 302 - Seção 302 da Sarbanes-Oxley	Alteração não autorizada de dados
Sarbanes-Oxley Section 404 - Seção 404 da Sarbanes-Oxley	A modificação de dados, acesso não autorizado
Sarbanes-Oxley Section 409 - Seção 409 da Sarbanes-Oxley	A negação de serviço, acesso não autorizado

Regulamento	**Ameaça potencial à segurança**
Gramm-Leach-Bliley - Gramm-Leach-Bliley	O acesso não autorizado, modificação ou divulgação
Health Insurance Portability and Accountability Act (HIPAA) 164.306 - Portabilidade de Seguros de Saúde e Accountability Act (HIPAA) 164,306	Acesso não autorizado a dados
HIPAA 164.312 HIPAA 164,312	Acesso não autorizado a dados
Basel II – Internal Risk Management - Basiléia II - Gestão de Riscos Internos	Acesso não autorizado a dados
CFR Part 11- CFR Part 11	Acesso não autorizado a dados
Japan Privacy Law - Japão Direito de Privacidade	Acesso não autorizado a dados
EU Directive on Privacy and Electronic Communications - Directiva da UE relativa à privacidade e às comunicações electrónicas	Acesso não autorizado a dados
Payment Card Industry Data Security Standard (PCI DSS) - Payment Card Industry Data Security Standard (PCI DSS)	Alteração não autorizada de dados

Este sistema permite criar políticas de segurança flexíveis para o banco de dados. Por exemplo, qualquer usuário do banco de dados, tais como SYSTEM, que tem o papel DBA, pode fazer modificações em parâmetros básicos de um banco de dados. Suponha que um administrador inexperiente, que tem privilégios de sistema, decida iniciar um novo arquivo de log, mas não percebe que fazendo isso em um determinado momento pode causar problemas para o banco de dados.

Com o Oracle Database Vault, você pode criar uma regra para evitar este comando do usuário de fazer essas modificações, limitando o seu uso do ALTER SYSTEM SWITCH LOGFILE (ORACLE, 2010). Esta ferramenta também permite anexar regras a regra de comando para restringir ainda mais a atividade, como limitar a execução da instrução.

A consolidação de banco de dados pode resultar em várias contas de usuário poderoso que residem em um único banco de dados. Isso significa que, além do banco de dados DBA geral, o esquema proprietário de aplicativos individuais também podem ter privilégios poderosos. Revogar alguns privilégios podem afetar negativamente os aplicativos existentes.

O Database Vault possui o sistema chamado Reinos, que permite o acesso às aplicações através de um caminho confiável, evitando que os usuários do banco de dados que não tenham sido especificamente autorizados tenham privilégios e acesso aos dados dos aplicativos. Por exemplo, um DBA que tem o privilégio SELECT ANY

TABLE pode ser impedido de utilizar esse privilégio para ler dados do aplicativo (ORACLE, 2010).

O Database Vault é contra o acesso não autorizado a dados de aplicativos, bem como a alterações no banco de dados realizadas por qualquer usuário, até mesmo por privilegiados como o DBA, de forma intencional ou acidental que possam de alguma forma serem danosas, levando em consideração vários fatores, como horário, autenticação, aplicativo e outros.

7.2. IBM InfoSphere Guardium

Os sistemas de banco de dados DB2 fornecem um mecanismo de auditoria para ajudar na detecção de acesso desconhecida ou imprevista aos dados. A instalação de auditoria DB2 gera e permite a manutenção de uma trilha de auditoria para uma série de eventos pré-definidos no banco de dados.

Para os sistemas de proteção de missão crítica, a maioria das organizações tem políticas de mudança formal de controle que determinam como e quando os empregados e prestadores de serviços podem fazer alterações nas bases de dados de produção. No entanto, é difícil detectar violações, fazendo com que as políticas sejam de difícil aplicação. Como solução, a IBM oferece a ferramenta Guardium InfoSphere, que promete enviar em tempo real, alertas de segurança sempre que forem feitas alterações importantes no sistema. Dentre as funcionalidades estão (IBM, 2010):

> • **Proteção de Fraude para Sistemas SAP:** Desde dados do cliente até ERP e informação pessoal, os sistemas SAP muitas vezes contêm informações sigilosas que precisam ser monitoradas para propósitos de regulação e auditorias. As empresas conseguem detectar fraudes em tempo real através do monitoramento de todas as atividades do utilizador incluindo atividades dos administradores e pessoal subcontratado. O InfoSphere Guardium fornece informação mais detalhada acerca dos utilizadores SAP, tornando mais fácil para as empresas detectar atividades fraudulentas sem executar quaisquer mudanças nas suas bases de dados ou aplicações.
>
> • **Proteção de ficheiros SharePoint:** Os repositórios SharePoint muitas vezes contêm informações sigilosas tais como os resultados financeiros da empresa ou propriedade intelectual valiosa, mas não existem os meios necessários para prevenir um uso incorreto por parte dos funcionários.
>
> • **Suporte para o Mainframe:** oferece capacidades reforçadas de monitorização da atividade verificada nas bases de dados IBM DB2 ao rodar o System Z, permitindo às empresas proteger informação crítica de acesso não autorizado por parte dos administradores. Por exemplo, se um administrador de uma base de dados numa companhia de seguros tentar obter o número de segurança social de um cliente, salário e histórico médico, o sistema irá imediatamente gerar um alerta para o pessoal de segurança e regulamentação. Além disso, as empresas podem realizar uma série de testes automatizados para avaliar vulnerabilidades de segurança, tais como permissões fracas que possam levar a organização a expor-se a perdas de dados ou falhas em auditorias de regulação.

> • **Melhoria de Regulação e Auditoria de processos:** o novo software permite que as empresas tenham mais flexibilidade para definir fluxos de trabalho personalizados e partilhar informação específica de auditoria com audiências relevantes nas suas organizações. Juntamente com o relatório de software pré-empacotado de regulações comuns tais como SOX, HIPAA e PCI, esta capacidade ajudará as empresas a poupar tempo e dinheiro ao reduzir significativamente o tempo necessário a reunir e relatar os dados de regulação requeridos pelos auditores.
>
> • **Bloqueio e Quarentena avançados:** As empresas podem bloquear seletivamente utilizadores individuais para não acessarem o sistema por um determinado período de tempo no caso de atividade suspeita ou não autorizada, evitando assim a perda de dados valiosos até que a atividade possa ser investigada. Por exemplo, se um administrador de uma base de dados num hospital acessar dados confidenciais de um paciente, o acesso deste funcionário será automaticamente bloqueado, sem ser necessária qualquer mudança manual, dispendiosa ou propensa ao erro das bases de dados e aplicações.

O InfoSphere Guardium permite a simplificação da regulação e segurança das organizações com um único conjunto de controles centralizados e automatizados para uma larga gama de aplicações e base de dados das empresas (IBM, 2010). Para além das suas capacidades de monitorização automatizada, o novo software ajuda os clientes a cumprir com a regulação mais facilmente ao fornecer controle mais preciso das informações, assegurando a privacidade e a integridade de dados corporativos e simplificando auditorias.

8. CONCLUSÕES

Este trabalho apresenta a importância de se garantir a segurança, inviolabilidade e integridade das informações contidas em um sistema informatizado de gestão. Fez-se um levantamento e descrição das principais pesquisas na área que propõem mecanismos para a proteção dos registros de auditoria. Com isto, conclui-se que com a dependência cada vez maior dos sistemas de armazenamento de dados críticos, deve-se desenvolver novas soluções para o monitoramento e proteção destes dados.

Atualmente, existem alguns estudos na área com algumas implementações, mas nenhum deles se apresentou como uma solução ótima para o problema de segurança de logs. Também foram apresentados dois sistemas comerciais da Oracle e da IBM que prometem resolver os problemas, como acessos indevidos a informações sigilosas e integridade dos registros de auditoria. Com base nesta investigação, percebe-se que esta é uma área muito promissora e importante de estudo, sendo recomendado a realização de novas pesquisas na área de controle e segurança da informação por meio da utilização de tecnologias de Business Intelligence.

REFERÊNCIAS

Bellare, Mihir; Yeey, S., Bennet. **Forward Integrity For Secure Audit Logs**. Dept. of Computer Science & Engineering, Mail Code 0114, University of California at San Diego, 1997.

Bosworth, Seymour; Kabay, M.E. **COMPUTER SECURITY HANDBOOK Fourth Edition.** John Wiley & Sons, Inc. 2002 Canada. ISBN 0-471-41258-9.p. 28-846.
Elmasri, Ramez; Navathe B. Shamkant. **FUNDAMENTALS OF DATABASE SYSTEMS 4th ed**. Copyright © 2004 by Pearson Education, Inc. ISBN 0-321-12226-7.p. 735.
Hawthorn, P., , B., Clifton, C., Wagner, D., Bellovin, S. M., Wright, R. N., Rosenthal, A., Poore, R. S., Coney, L., Gellman, R., e Hochheiser, H. (2006). **Statewide databases of registered voters: a study of accuracy, privacy, usability, security, and reliability issues**. Communications of the ACM, 49(4):26–28.

Holt, E.,Jason. **Logcrypt: forward security and public verification for secure audit logs**. Internet Security Research Lab, Brigham Young University. ACSW Frontiers '06 Proceedings of the 2006 Australasian workshops on Grid computing and e-research - Volume 54.

IBM. **IBM InfoSphere Guardium**. Encontrado em: http://www-01.ibm.com/software/data/guardium/, na data de: 20/12/2010.

Jangra, A.; Bishla, D.; Bhatia, Komal; Priyanka. **Functionality and Security Analysis of ORACLE, IBM-DB2 & SQL Server.** Global Journal of Computer Science and Technology. Vol. 10 Issue 7 Ver. 1.0 September 2010. p. 8.
MCDOWALL, R.D. **Validation of Spectrometry Software — Audit Trails for Spectrometer Software**. Spectroscopy 22(4) April 2007. p.16-18.
http://spectroscopyonline.findanalytichem.com/spectroscopy/data/articlestandard/spectrscopy/172007/421873/article.pdf.

Menezes J. Alfred; Van Oorschot C. Paul; Vanstone A. Scott.**HANDBOOK of APPLIED CRYPTOGRAPHY**. Massachusetts Institute of Technology June 1996. p.560.

Oracle. Apresentação do Oracle Database Vault. Encontrado em: http://download.oracle.com/docs/cd/B28359_01/server.111/b31222/dvintro.htm&prev=_t&rurl=translate.google.com.br&twu=1&usg=ALkJrhjnhkUAHhpz2vIjNKVO8sXgsNt0kw#CEGCIECD, na data de 10/11/2010.
Peterson, N. J. Zachary; Burns Randal; Ateniese, Giuseppe; BONO Stephen. **Design and Implementation of Verifiable Audit Trails for a Versioning File System**.
Proceeding FAST '07 Proceedings of the 5th USENIX conference on File and Storage Technologies 2007.
Sallachl, D. L. (1992).**A deductive database audit trail**. In Proceedings of the 1992.acm/SIGAPP Symposium on Applied computing (SAC'92), p. 314–319.

Schneier, Bruce; Kelsey, John. **"Secure audit logs to support computer forensics"**. ACM Transactions on Information and System Security, 2(2), 1999. p.159-176.
Simon, Fernando; Dos Santos, L., Aldri; Hara S. Carmem. **Um Sistema de Auditoria baseado na Análise de Registros de Log**. Departamento de Informática Universidade Federal do Paraná (UFPR). Escola Regional de Banco de Dados (ERBD'2008), Florianópolis-SC, Abril 2008.

Waters R. Brent; Balfanz, Dirk; Durfee, Glenn; Smetters, D. K. **Building an Encrypted and Searchable Audit Log**. CiteSeerX - Scientific Literature Digital Library and Search Engine (United States). 2004.

Xu, Wensheng; Chadwick, David; Otenko Sassa. **A PKI Based Secure Audit Web**. In *IASTED Communications, Network and Information and CNIS*, Phoenix, USA, November 2005. Encontrado em: http://www.oracle.com/global/br/corporate/press/2008_mar/Oracle_Database_Vault.html

APPENDIX 1

Table 1. MCDA methods (Adapted from Guitouni and Martel (1998))

No	Method	Author(s)
	Linear weighting and elementary methods	
1	Weighted Sum	Churchman and Ackoff (1954)
2	Lexicographic method	Roy and Hugonnard (1982)
3	Conjunctive method	Hwang and Youn (1981)
4	Disjunctive method	Chen and Hwang (1992)
5	Maximin method	Hwang and Youn (1981)
	Single synthesizing criterion or utility theory	
6	TOPSIS	Hwang and Youn (1981)
7	MAVT	Keeney and Raifa (1976)
8	UTA	Jacquet-Lagreze and Siskos (1982)
9	SMART	Edwards (1971)
10	MAUT	Bunn (1984)
11	AHP and ANP	Saaty (1980), Saaty (2005)
12	DEA	Talluri et al. (1999)
13	COPRAS	Zavadskas et al. (2007); Chatterjee et al. (2011)
	Outranking methods	
14	ELECTRE	De Boer et al. (1998); Dulmin and Mininno (2003)
15	ELECTRE I	Roy (1968)
16	ELECTRE IS	Roy and Bouyssou (1993)
17	ELECTRE II	Roy and Bertier (1971)
18	ELECTRE III	Roy (1978)
19	ELECTRE IV	Roy and Hugonnard (1982)
20	ELECTRE TRI	Yu (1992); Mousseau et al. (2000)
21	PR OMETHEE	Dulmin and Mininno (2003)
22	PROMETHEE TRI	Figueira et al. (2004)
23	PROMETHEE/GAIA technique	Dulmin and Mininno (2003)
24	NAIADE	Munda (1995)
25	ELECCALC	Kiss et al. (1994)
26	UTADIS	Doumpos et al. (2001)
27	MELCHIOR	Leclerc (1984)
28	ORESTE	Roubens (1980)
29	REGIME	Hinloopen and Nijkamp (1982)
30	PROMSORT	Araz and Ozkarahan (2007)
31	EVAMIX	Voogd (1983)
32	QUALIFLEX	Paelinck (1978)
	Fuzzy methods	
33	Fuzzy relationship hierarchy	Lin and Chen (2004)
34	Fuzzy set approach	Sarkar and Mohapatra (2006)
35	Fuzzy suitability index (FSI)	Bevilacqua et al. (2006)
36	Fuzzy weighted sum	Baas and Kwakernaak (1977)
37	Fuzzy miximini	Bellman and Zadeh (1970)
38	AI methods	Ng and Skitmore (1995); Vokurka et al. (1996); Kwong et al. (2002); Choy et al. (2002); Choy et al. (2003); Choy et al. (2005)
39	CBR	Ng and Skitmore (1995); Choy et al. (2003)
	Mixed methods	
40	Martel and Zaras method	Martel and Zaras (1990); Martel and Zaras (1995)
41	Fuzzy conjunctive/ disjunctive method	Dubois, Prade and Testemale (1988)

APPENDIX 2

Operational requirements identified for the purpose of criteria-based evaluation are as follows:

1. The maximum tender for the evaluation is MVR1.500,000.00.
2. The minimum tender for the evaluation is MVR25.000.00.
3. Different cost bands are evaluated differently.
4. Public announcement should be made for every procurement costing more than MVR25.000.00.
5. There is a minimum of two criteria for evaluation.
6. There can be more criteria for evaluation based on the procurement.
7. Allocation of criteria and weights are based on the needs of the organisation.
8. A pre-bid meeting is compulsory and it needs to be announced.
9. Specification should be provided to potential bidders during the pre-bid meeting.
10. Marking criteria with weights are provided in advance in pre-bid meeting.
11. All required documents should be submitted with the bid and the requirements need to be informed to bidders.
12. If any bidder requires, calculations procedures are explained.
13. All bids are submitted on specific date and time. All the documents are checked verified during the submission process.
14. It requires minimum three BEC members to evaluate bids.
15. Basis for evaluation solely depends on the information provided in pre-bid meeting.
16. Suppliers' bids need to be verified for correct information.
17. Suppliers' previous jobs are evaluated based on available information.
18. Submitted support documents are primary source of information and they are assessed.
19. Assess the bid price compare to the expected work.
20. Suppliers' performances are evaluated based on the criteria provided and according to the weights and marks in allocated schemes provided in advance.
21. Marks are allocated based on the criteria and weights provided during pre-bid meeting in relation to performances of suppliers.
22. Technical expertise is used to get advice and explanations on procurement of technical good and services.
23. A through check is made if the proposed goods or services meet the specified standard.
24. Every criterion is assessed independently from one another.
25. All the criteria need to be evaluated.
26. No ranking can be made in evaluation; rather, marks are allocated in evaluation.
27. Pair wise comparison cannot be done.
28. In evaluation stage no changes to criteria, weights and requirements should be made.
29. Incomplete bids should be rejected.
30. Evaluation calculations are shown to bidders if requested.
31. BEC needs to approve of the winner. Evaluation analysis does not grant awarding the bid to the winner.
32. BEC need to state the reason for selection of the specific bid.
33. Bidders are informed the winner but not marks.

34. If any bidder wants more clarification, evaluation calculations are shown.
35. No discrimination in evaluation.
36. Evaluation method needs to be accurate.
37. Evaluation method should be using reasonable amount of resources and provide reasonable results.
38. Evaluation method should comply with procurement rules and regulations.
39. Evaluation method should provide no chance of manipulation from both sides.
40. Evaluation method needs to help minimise complaints.
41. Evaluation method needs to support utility concept.
42. Evaluation method should be clear and easily understandable.