

Como manipular usuários do Firebird via Front-End?

Essa é, com certeza, uma das dúvidas mais freqüentes em fóruns e listas de Firebird/Delphi. São muitos os desenvolvedores que querem dar autonomia de gerenciamento de usuários ao seu sistema. Apesar de bastante discutida, a questão sempre retorna como se as dúvidas em torno desta permanecessem ávidas.

Este artigo não é uma receita de bolo ou uma solução definitiva, mas sim fruto de algumas pesquisas e experiências a fim de mostrar algumas alternativas disponíveis para o contexto. Cabe ao leitor decidir qual, quando e como usar. Apegarei-me ao Delphi, por ser o mais usado, mas muitas das soluções podem ser portadas pra outro front-end.

Como segurança é o lema, sugiro a leitura do artigo do Ivan Prenosil "[Segurança - ISC4.gdb otimizado](#)", que está na seção de artigos traduzidos da CFLP. Um detalhe a se levar em conta é que a partir do Firebird 1.5, a security.fdb (antes isc4.gdb) é acessada pelo motor através de transações read-only. Portanto, nem tudo deste artigo citado acima poderá ser aplicado a esse caso.

Então, vamos lá...mãos à obra!!!

Alternativa 1: MDO Admin (Mercury Database Objects)

A suíte Mercury Database Objects (MDO) é fruto da nossa terra! Desenvolvida por nós, brasileiros, foi inspirada na suíte FreeIBComponents e Interbase Express 4.3. Sob a coordenação de Henrique Meira, a suíte que se encontra na versão 0.8.8 (até a data de confecção deste artigo), está rompendo fronteiras e ganhando âmbito internacional, obtendo diversos colaboradores de vários países. Pra quem deseja usar uma suíte semelhante a IBX, mas que garanta compatibilidade com futuras versões do Firebird, MDO é uma ótima pedida. Disponível em <http://sourceforge.net/projects/mdo>.

A MDO apresenta um conjunto de componentes voltados à administração de bases Firebird (Backup, Segurança, Informações da Base, etc), disponíveis na Paleta MDO Admin.

O componente da paleta MDO Admin que nos interessa neste artigo chama-se



MDOSecurityService

Vejamos apenas as propriedades e métodos que mais nos interessam.

Propriedades:

- **UserName:** Bastante intuitivo, não? Aqui tanto se pode definir ou buscar o User Name do usuário em questão;
- **Password:** Define a senha do usuário ao adicionar ou modificar as informações do mesmo;
- **SQLRole:** Define-se ou obtém o role usado pelo usuário ao se conectar ao isc4.gdb;
- **UserInfo:** Esta propriedade é um array do tipo Record e armazena todas as informações dos usuários do isc4.gdb;
- **UserInfoCount:** Retorna o nº de elementos de TUserInfo.
- **FirstName, MiddleName, LastName, UserID e GroupID:** Estas propriedades preenchem dados opcionais de usuários (Primeiro nome, nome do meio, sobrenome, UID e GID).

Os métodos para adicionar, deletar, modificar, exibir apenas um e exibir todos os usuários são, respectivamente, AddUser, DeleteUser, ModifyUser, DisplayUser e DisplayUsers.

MDOAdmin não tem muito segredo. Uma boa “fuçada” e alguns testes são suficientes para sua utilização. De qualquer forma, disponibilizei um simples sample que exemplifica o uso do MDOSecurityService.

Alternativa 2: InterBase Admin (IBX)

InterBase Admin é uma suíte de componentes que se utilizam de chamadas à API do Firebird/Interbase para realizar operações de administração de bases de dados como configurações, Backup/Restore, Validação, Estatísticas, Logs e, entre outros, Segurança, que é onde vamos nos ater. IBX será uma opção apenas enquanto o Firebird ainda mantiver compatibilidades com o InterBase, o que sabemos que não irá durar muito :-). Como MDO foi baseada no IBX, o que foi dito a respeito do funcionamento da MDO Admin serve para o IB Admin.

Alternativa 2: IBO Admin

IBO Admin é uma solução desenvolvida por Lorenzo Mengoni e pode ser encontrada em www.mengoni.it/downloads.html.

É uma suíte de gerenciamento de usuários muito eficiente e bastante semelhante ao InterBase Admin, com versões para Delphi e C++Builder.

Seu uso, propriedades e métodos são semelhantes aos das Suítes MDO e IBX , mas , assim como MDOAdmin, IBOAdmin garante compatibilidade com futuras versões do Firebird.

Alternativa 3: GSEC

O GSEC é um utilitário de linha de comando para gerenciar usuários no FB e é instalado junto com este, no diretório “bin”.

A vantagem do GSEC é que ele não se limita a uma linguagem ou um ambiente de programação e pode ser usado até em Shell Scripts (Linux) ou arquivos de lote (Windows).

No Windows, é necessário informar o login e senha do Administrador (SYSDBA), já no Linux, pode ser feito diretamente e como usa o usuário ativo no sistema, deve ter as devidas permissões.

gsec [<options>] <command> [<parameter> ...]

Sendo que **options** pode ser:

- user <login do administrador>
- password <senha do administrador>
- role <SQL Role do administrador>
- database <local do isc4.gdb ou security.gdb>

command pode ser:

1 – adicionando usuário:

- add <name>

Windows: gsec -user sysdba -password masterkey -add usuario -pw senha

Linux: #gsec -add usuario -pw senha

2 – excluindo usuário:

-delete <name>

Windows: gsec -user sysdba -password masterkey -delete usuario

Linux: #gsec -delete usuario

3 – modificando usuário:

-modify <name> <parameter> [<parameter>]

Windows: gsec -user sysdba -password masterkey -modify usuario -pw nova_senha

Linux: #gsec -modify usuario -pw nova_senha

4 – exibindo dados de um usuário:

-display <name>

Windows: gsec -user sysdba -password masterkey -display usuário

Linux: #gsec -display usuario

5 – exibindo dados de todos os usuários:

-display

Windows: gsec -user sysdba -password masterkey -display

Linux: #gsec -display

Parâmetros disponíveis:

-pw <password>

-uid <uid>

-gid <gid>

-fname <firstname>
-mname <middlename>
-lname <lastname>

Nota: O GSEC pode se tornar um grande problema se o acesso ao mesmo for dado à pessoa errada.

Alternativa 4: TIB_Connection – Suíte IB *Objects*

Na suíte IB *Objects*, de Jason Wharton, o componente TIB_Connection e seus descendentes possuem o método AlterUser, que permite adicionar, modificar e excluir usuários do Firebird. Até a versão 4.3.Aa não é possível exibir dados de usuários cadastrados. Talvez, futuramente isso seja implementado.

A sintaxe do AlterUser é:

AlterUser(Ação, UserName, Password, GroupName, FirstName, MiddleName, LastName);

Sendo que a ação pode ser uaAddUser, uaDeleteUser ou uaModifyUser, para adicionar, deletar e modificar, respectivamente.

Ex: **AlterUser**(uaAddUser, 'HUGO_BERETA', 'naosei', '', 'Hugo', 'de Freitas', 'Bereta');

Alternativa 5: UDF's e DLL's

- A rFunc, encontrada em <http://sourceforge.net/projects/rfunc> é uma UDF escrita em C++, que além de ter inúmeras funções para manipulação de strings, valores numéricos, Blobs e operações com bits, ainda permite fazer o controle de usuários diretamente pelo Banco.
- Para aqueles que gostam de tomar as rédeas e criar sua própria tabela de usuários, por que não usar a mesma criptografia que Firebird usa com a security.fdb/isc4.gdb? A UnixPass, desenvolvida por Gérard Perreault e encontrada no site de Dunstan Thomas, <http://interbase.dthomas.co.uk/>, possibilita o uso de tal criptografia. No mesmo site, é possível encontrar exemplos de uso da unixpass. Um uso simples e conhecido é o de se criptografar a senha na hora de grava-la na base, e, no login do usuário, compará-la com o que foi digitado e criptografado pelo mesmo processo.

- Ainda no site do Dunstan Thomas, encontra-se uma UDF (IBUsers.zip) enviada por Paul Beach, Inprise Europe, que permite fazer o controle de usuários, além de backup da base de dados. A idéia de Paul consiste em ter um BD à parte para utilizar a UDF, que, por sua vez, se utiliza do GSEC e GBAK através de Stored Procedures para as operações de controle de usuários e Backup, respectivamente. Os usuários são manipulados na base de usuários padrão do Firebird (security.fdb/isc4.gdb).

Bem, por enquanto é só! Acho que já dá pra escolher uma das alternativas e manter os usuários de seus sistemas na linha. Mas, lembre-se que toda esta paranóia aqui não serve pra nada se não se levar em conta a segurança do servidor como um todo. Se este não lhe oferecer segurança ou mesmo se você permitir que qualquer um tenha acesso a ele, se alguém substituir sua base de usuários por uma outra da qual ele tenha as senhas, já era toda a sua paranóia em cima de sua base de dados.

<p>Artigo Original</p> <p>Hugo Bereta (Colaborador da CFLP)</p> <p>hugo.bereta@comunidade-firebird.org</p>	<div data-bbox="1065 877 1243 1058" data-label="Image"> </div> <p>Comunidade Firebird de Língua Portuguesa</p> <p>Visite a Comunidade em:</p> <p>http://www.comunidade-firebird.org</p>
<p>A Comunidade Firebird de Língua Portuguesa foi autorizada pelo Autor do Original para divulgar este trabalho</p>	