We Translate
Business Processes

from the Mind
to the Computer
to the Bottom Line.

BUSINESS & COMPUTERS, Inc.
13839 Mur-Len Rd, Suite M
OLATHE, KANSAS 66062

Phone: (913) 764-2311
Fax: 764 7515
larryg@kcnet.com

SQL 7/2000 Security for In House Networks

Copyright® 2002 Business & Computers, Inc.

A note – the below is my humble opinion – with testing – If you use my ideas please test them and if you have problems or learn more let me know.

SA is:

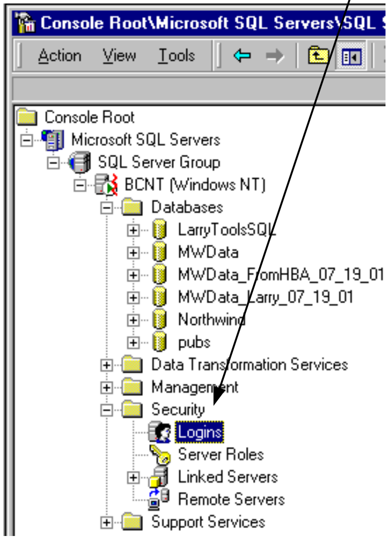
per SQL Server

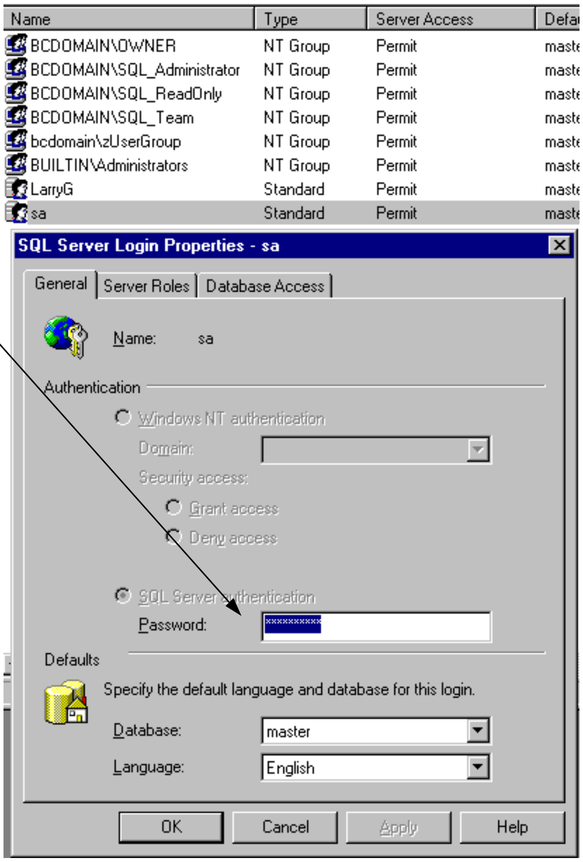
Logging In

I - Does SA Stand for "Suddenly Axed?" (This only applies to Mixed Mode see Section II)

A) **If you do not have a password for SA (System Administrator) you DO NOT HAVE security on your server and your database.** (It's hard not to set a password in SQL Server 2000)

1) Try to log into your server using sa with no password. If it works, anyone can get into the server and have complete rights to see, edit, add, and delete your data, and full access to all objects in your database. (You don't mind if Suzie was playing around in your database and deleted the invoice table. Do You?)
2) To put a password on sa, go to Security, Logins, and then right click on sa, enter your password.
3) Please do the above before Suzie does. It might be hard to keep your job, if no one can log into the server because no one knows Suzie's password.





Name	Type	Server Access	Defa
BCDOMAIN\OWNER	NT Group	Permit	maste
BCDOMAIN\SQL_Administrator	NT Group	Permit	maste
BCDOMAIN\SQL_ReadOnly	NT Group	Permit	maste
BCDOMAIN\SQL_Team	NT Group	Permit	maste
bcdomain\zUserGroup	NT Group	Permit	maste
BUILTIN\Administrators	NT Group	Permit	maste
LarryG	Standard	Permit	maste
sa	Standard	Permit	maste

SQL Server Login Properties - sa

General | Server Roles | Database Access

Name: sa

Authentication

○ Windows NT authentication
Domain:
Security access:
○ Grant access
○ Deny access

● SQL Server authentication
Password: **********

Defaults

Specify the default language and database for this login.

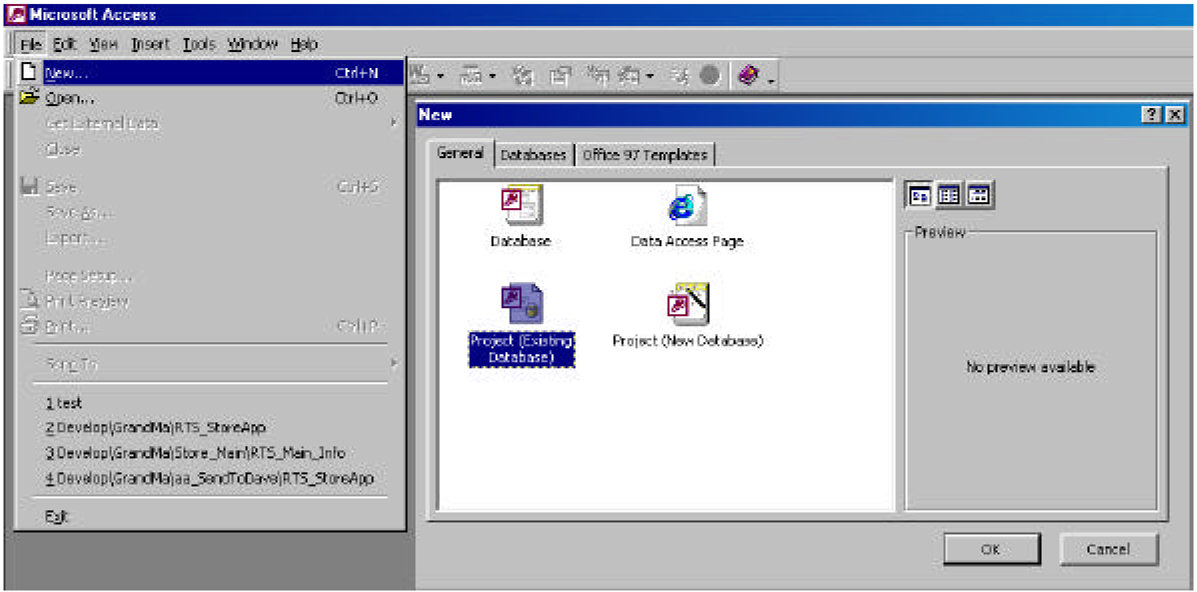
Database: master
Language: English

B) Did you think that if you used NT Authentication and connected to your SQL Server in code from the desk tops, you didn’t have to worry about a password for sa?

1) You say you don’t have to worry, because my users can’t get to the server and they don’t have the tools to look at my objects on the server.

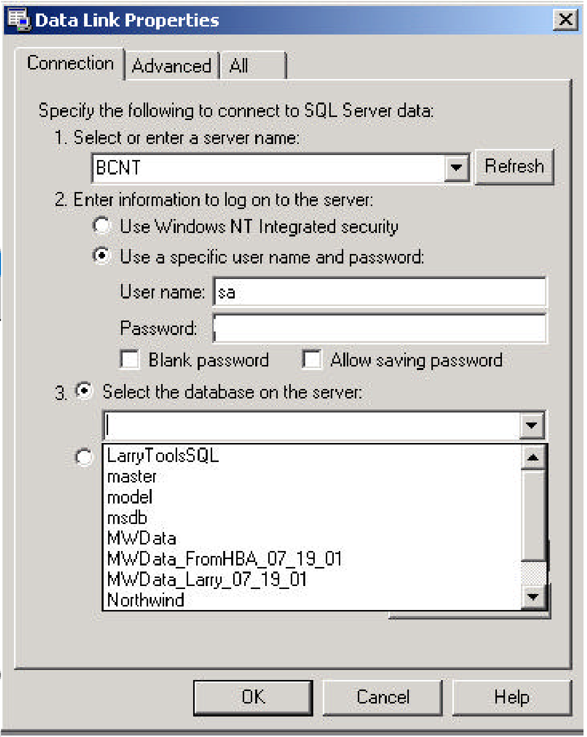
2) **Do they have MS Access on their machines?**

3) If they do, and sa has no password, they can get to your database and all the objects.
* On Microsoft Access 2000 go File/New/Project(Existing Database)



* The Data Link Properties come up. Pick or type in your server, type sa for user name, put a checkmark in blank password, and pick your database.

* You are now the master of the database. You can add/ edit/delete data, and you can add/edit or delete objects.

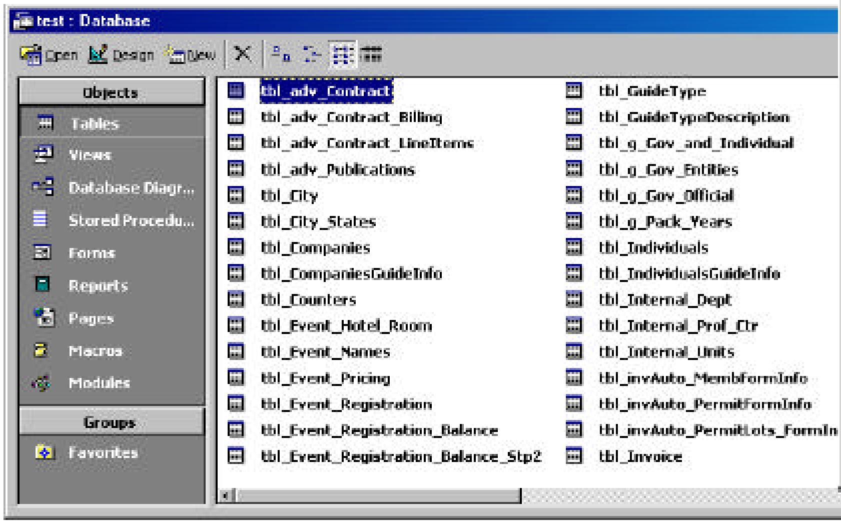


Specify the following to connect to SQL Server data:

1. Select or enter a server name:
BCNT

2. Enter information to log on to the server:
○ Use Windows NT Integrated security
● Use a specific user name and password:
User name: sa
Password:
□ Blank password □ Allow saving password

3. ● Select the database on the server:
○
LarryToolsSQL
master
model
msdb
MWData
MWData_FromHBA_07_19_01
MWData_Larry_07_19_01
Northwind



tbl_adv_Contract
tbl_adv_Contract_Billing
tbl_adv_Contract_LineItems
tbl_adv_Publications
tbl_City
tbl_City_States
tbl_Companies
tbl_CompaniesGuideInfo
tbl_Counters
tbl_Event_Hotel_Room
tbl_Event_Names
tbl_Event_Pricing
tbl_Event_Registration
tbl_Event_Registration_Balance
tbl_Event_Registration_Balance_Stp2
tbl_GuideType
tbl_GuideTypeDescription
tbl_g_Gov_and_Individual
tbl_g_Gov_Entities
tbl_g_Gov_Official
tbl_g_Pack_Years
tbl_Individuals
tbl_IndividualsGuideInfo
tbl_Internal_Dept
tbl_Internal_Prof_Ctr
tbl_Internal_Units
tbl_invAuto_MembFormInfo
tbl_invAuto_PermitFormInfo
tbl_invAuto_PermitLots_FormIn
tbl_Invoice

C) If you do have a password for SA you are on your way to a secure database.

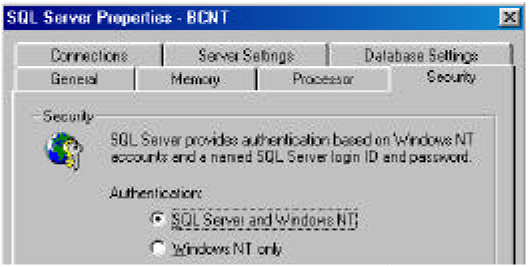
The Process at a Glance

Windows NT = NT Windows 2000= Win 2K SQL 7 SQL 2000 = SQL2K

Job	System	How to do it
1) Create Groups on your network and then add users to your groups. Such as SQL_ReadOnly SQL_Team SQL_Adminstrator SQL_AccRec SQL_AccPayable Etc.	NT	Programs/Administrative Tools/User Manager for Domains. User/New Local Group
	2000	Control Panel then Administrative Tools then Computer Management then Local Users & Groups
2) Add the groups to SQL Server. This can be done from the Query Analyzer OR The Enterprise Manager Either way you do this process, you will need to pick the databases the group has permission to login into. You would do the process at the live data site, bring a backup to your site and then add the groups from your site the same way.	Query Analyzer SQL2K SQL 7	Sp_grantlogin ‘BCDomain\SQL_Team’
	Enterprise Mgr	Go to Security, then right click on Logins, then pick New Login and follow the instructions below.
	SQL 7	Pick the Windows Domain first, then in the Name field, type in the group name to the right of the domain name. E.g. BCDomain\SQL_Team
	SQL2K	Click the button to the right of the name and pick the Group from the list.
3) Once you finish adding the above logins, you will see all the groups you added from the server you are on. (You will not see any from Groups you added from another site.)		
4) Add the Groups to your database (the actual database inside SQL Server)		In Enterprise Manager go the database in question. Go to Users, right click on Users and click New Database User… Then pick you Group from the list.
Just some Info -->	SQL 7	Inside Users, you will see all the groups you added from the server you are on. (You will not see any from Groups you added from another site.)
	SQL2K	Inside Users, you will see all the groups you added from the server you are on, plus the Groups you added from another site.
5) Add Roles to your database. Such as RL_Admin RL_ReadOnly RL_Team RL AccPayable		In Enterprise Manager go the database in question. Go to Roles, right click on Roles and click New Database Role and name the role. Add the groups to the role and then sent the permission on individual objects. You should see the groups you added from your server and the groups you added from other servers

II - Users (NT) - Groups (NT) - Logins (SQL Server) - and - Roles (SQL Server).

*** Users, Groups, logins, and Roles, are a very important part of SQL Server Security. There are basically 2 types of security, also referred to as authentication, "SQL Server Security and Windows NT" <-(Win 95/98 machines must use) and "Windows NT only" (also known as "NT Integrated Security). I use the combination on my desktop & Laptop Servers, so I can connect to the SQL Server database on my Win 98 laptop when it is not connected to the NT network. From my users point of view, I use "Windows NT only", which means once they log into NT, their security is setup for SQL Server.**



Security

SQL Server provides authentication based on Windows NT accounts and a named SQL Server login ID and password.

Authentication:

● SQL Server and Windows NT

○ Windows NT only

Authentication is:

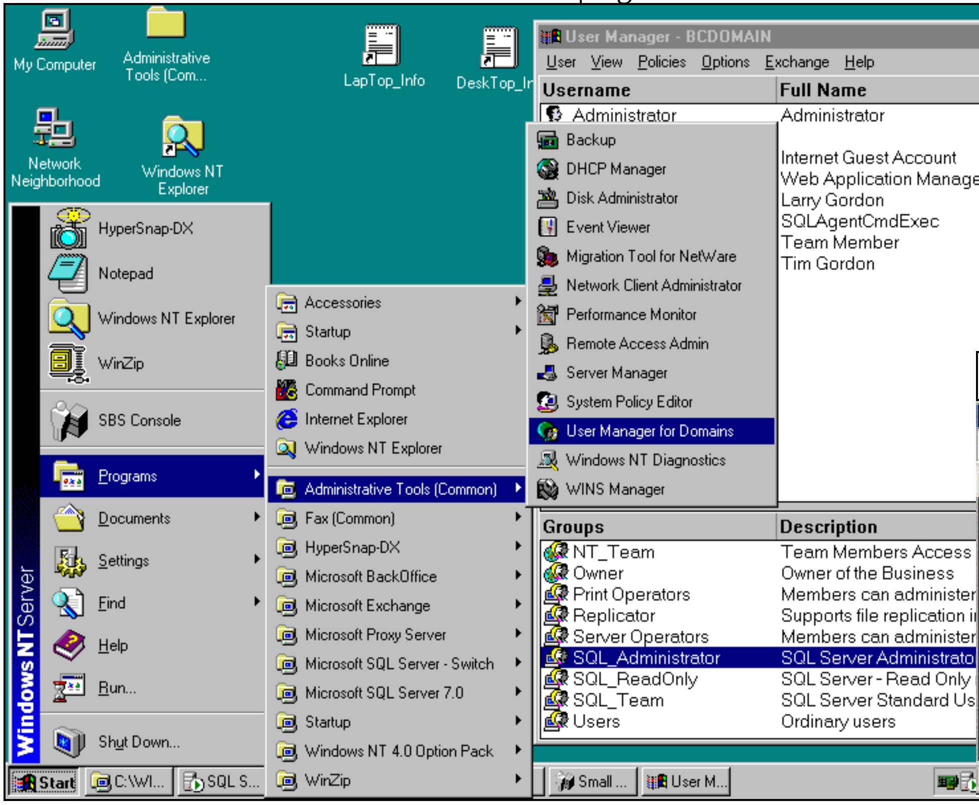
per SQL Server

A) NT Integrated Security (Usually referred to as NT Authentication or Trusted Connection**)**

1) Set up Groups on your NT/2000 machine that host's your SQL Server. Go to "User Manager for Domains" in NT (*See figure A-1)* or "Computer Management" in 2000 (*See figure A-2)*, then go to "User" on the menu and add "New Local Group" in Nt. You might want to setup the following groups.

SQL_ReadOnly
SQL_Team
SQL_Adminstrator
SQL_AccRec
SQL_AccPayable
Etc.

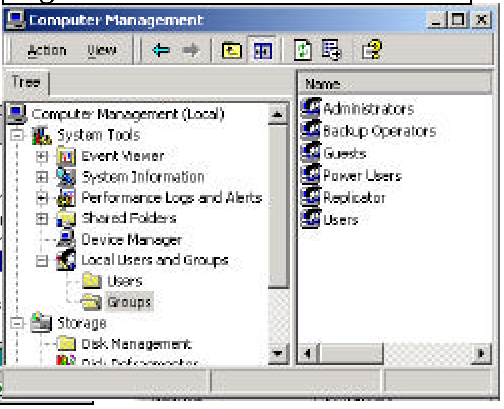
Figure A-1 Windows NT Server

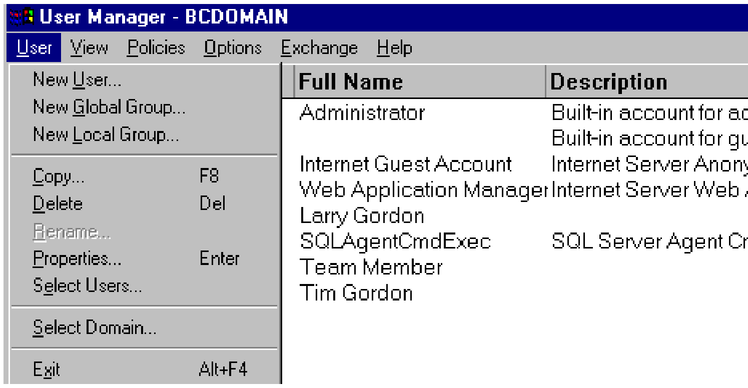


Username	Full Name	Descripti
Administrator	Administrator	Built-in ac
		Built-in ac
	Internet Guest Account	Internet Se
	Web Application Manager	Internet Se
	Larry Gordon	
	SQLAgentCmdExec	SQL Serv
	Team Member	
	Tim Gordon	

Groups	Description
NT_Team	Team Members Access
Owner	Owner of the Business
Print Operators	Members can administer
Replicator	Supports file replication i
Server Operators	Members can administer
SQL_Administrator	SQL Server Administrato
SQL_ReadOnly	SQL Server - Read Only
SQL_Team	SQL Server Standard Us
Users	Ordinary users

Figure A-2 Windows 2000 Server





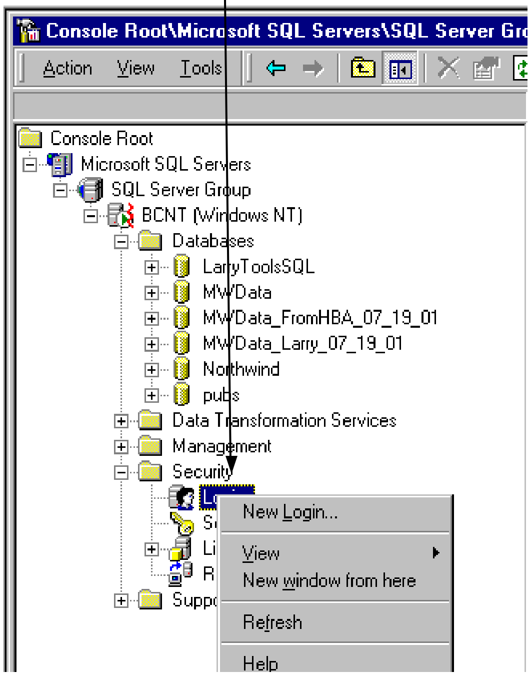
Full Name	Description
Administrator	Built-in account for ad
	Built-in account for gu
Internet Guest Account	Internet Server Anony
Web Application Manager	Internet Server Web /
Larry Gordon	
SQLAgentCmdExec	SQL Server Agent Cr
Team Member	
Tim Gordon	

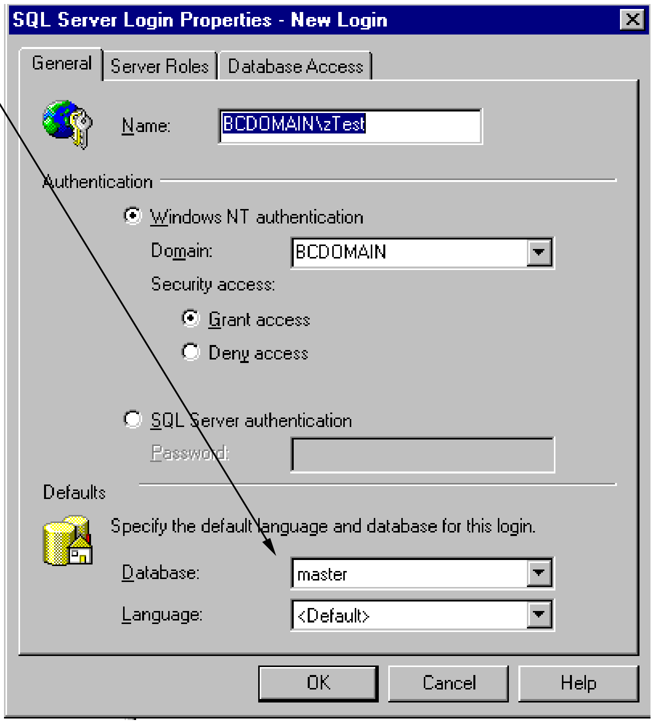
2) I recommend you use different groups than the NT server uses. That way you will have better control for SQL Server.

3) Then add the appropriate people to the groups. Double click on the group, click the add button, and add the people that should be part of the group.

SQL_ReadOnly
SQL_Team
SQL_Adminstrator
SQL_AccRec
SQL_AccPayable
Etc.

4) Go Into SQL Server Enterprise Manager, go to Security, then Login. Right click on Login, then pick New Login.
On the General tab of the form that comes up
a) Pick "Windows NT Authentication", Grant Access, and then pick your server.
b) Type in the name of the Group in the Name field next to your server.
c) Pick the default database for your user.




SQL Server Login Properties - New Login

Name: BCDOMAIN\zTest

Authentication

● Windows NT authentication

Domain: BCDOMAIN

Security access:

● Grant access

○ Deny access

○ SQL Server authentication

Password:

Defaults

Specify the default language and database for this login.

Database: master

Language: <Default>

On the Database Access tab

a) Pick the databases the user needs access to.

b) Pick "Public" only as the role for each database. (Public is a required role) (We will talk more about "Roles" later.)

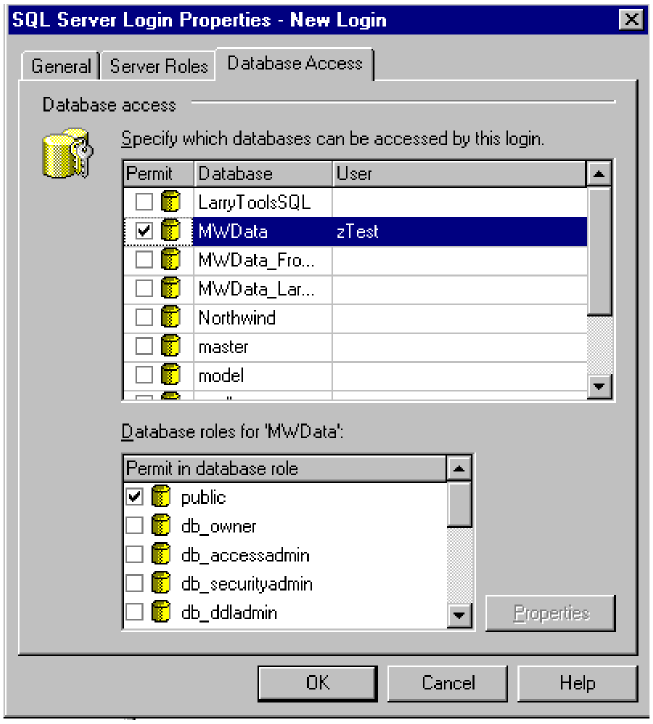
c) Just because you give a user, group, or role the right to login to a database they still can't use the objects in the database until you explicitly give them permission to the objects.

I recommend only using the "Public" database role for your database.

(You have no choice - if they are to get to your database they must be a member of public role.)

On the Server Roles, I recommend using only "System Administrators" role. This role is for any one who has all rights.

Look in Books On Line if you are interested in the other Server and Database roles furnished by SQL Server.



SQL Server Login Properties - New Login

General | Server Roles | Database Access

Database access

Specify which databases can be accessed by this login.

Permit	Database	User
□	LarryToolsSQL	
☑	MWData	zTest
□	MWData_Fro...	
□	MWData_Lar...	
□	Northwind	
□	master	
□	model	

Database roles for 'MWData':

Permit in database role
☑ public
□ db_owner
□ db_accessadmin
□ db_securityadmin
□ db_ddladmin

Authentication is:

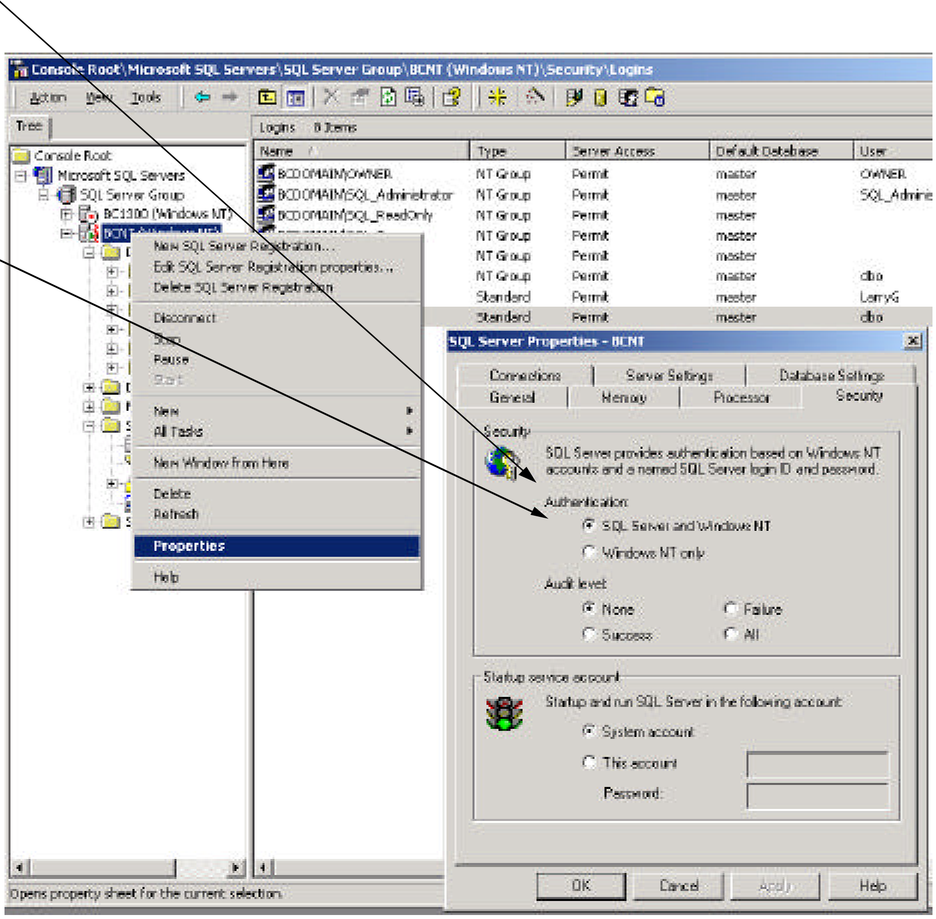
per SQL Server

B) SQL Server Security (SQL Server Authentication)

1) To see what type of security you are using with your server, right click on your server, go to properties, and then the security tab.

2) SQL Server Authentication is used for
   * The internet
   * Installations on Win 95/98/

Continued on the next page.



Console Root\Microsoft SQL Servers\SQL Server Group\BCNT (Windows NT)\Security\Logins

Logins 8 Items

Name	Type	Server Access	Default Database	User
BCDOMAIN\OWNER	NT Group	Permit	master	OWNER
BCDOMAIN\SQL_Administrator	NT Group	Permit	master	SQL_Admin
BCDOMAIN\SQL_ReadOnly	NT Group	Permit	master	
	NT Group	Permit	master	
	NT Group	Permit	master	
	NT Group	Permit	master	dbo
	Standard	Permit	master	LarryG
	Standard	Permit	master	dbo

SQL Server Properties - BCNT

Connections | Server Settings | Database Settings
General | Memory | Processor | Security

Security

SQL Server provides authentication based on Windows NT accounts and a named SQL Server login ID and password.

Authentication:

● SQL Server and Windows NT

○ Windows NT only

Audit level

● None ○ Failure

○ Success ○ All

Startup service account

Startup and run SQL Server in the following account:

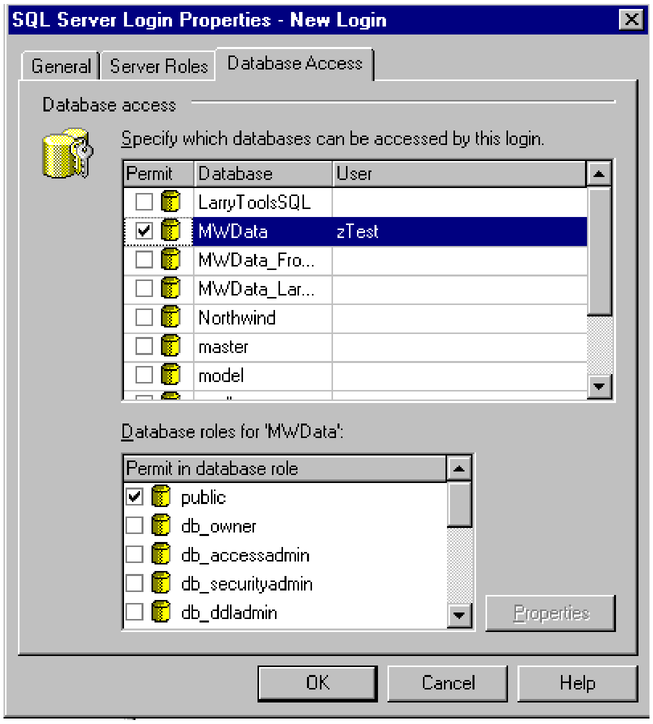
● System account

○ This account

Password:

Opens property sheet for the current selection.

3) To add a user (Called Login) to SQL Server, go into SQL Enterprise Manager and do one of the following.
a) If you only want to add the person for a particular database, right click on users under the database and pick “New Database User” and add the user.
b) If you want to add the person to multiple databases, go to Security, and right click on New Login. and add the user.
4) Adding the user is similar to adding a group, but you need to pick SQL Server authentication.
5) Don’t forget to add a password and pick the database(s) that the user has access to.


SQL Server Login Properties - New Login

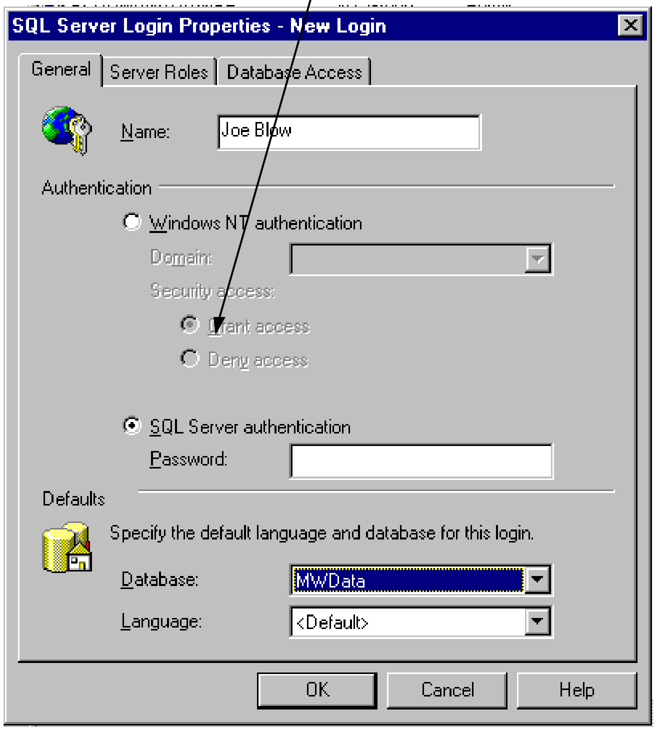
Database access

Specify which databases can be accessed by this login.

Permit	Database	User
□	LarryToolsSQL	
☑	MWData	zTest
□	MWData_Fro...	
□	MWData_Lar...	
□	Northwind	
□	master	
□	model	

Database roles for 'MWData':

Permit in database role
☑ public
□ db_owner
□ db_accessadmin
□ db_securityadmin
□ db_ddladmin


SQL Server Login Properties - New Login

Name: Joe Blow

Authentication

○ Windows NT authentication

Domain:

Security access:

● Grant access

○ Deny access

● SQL Server authentication

Password:

Defaults

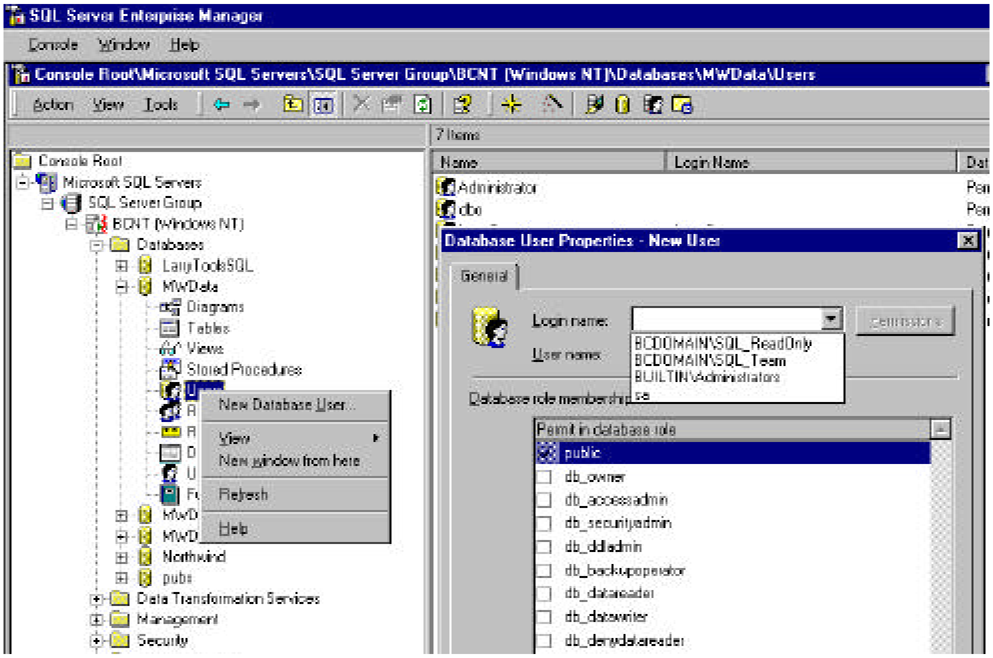
Specify the default language and database for this login.

Database: MWData

Language: <Default>

6) You can also add Users or Groups under “Users” inside your database.

User’s are
per
SQL Server


SQL Server Enterprise Manager

Name	Login Name	Dat
Administrator		Per
dbo		Per

Database User Properties - New User

Login name:

BCDOMAIN\SQL_ReadOnly
BCDOMAIN\SQL_Team
BUILTIN\Administrators
sa

User name:

Database role membership:

Permit in database role
☑ public
□ db_owner
□ db_accessadmin
□ db_securityadmin
□ db_ddladmin
□ db_backupoperator
□ db_datareader
□ db_datawriter
□ db_denydatareader

C) SQL Roles

Roles are:
per Database

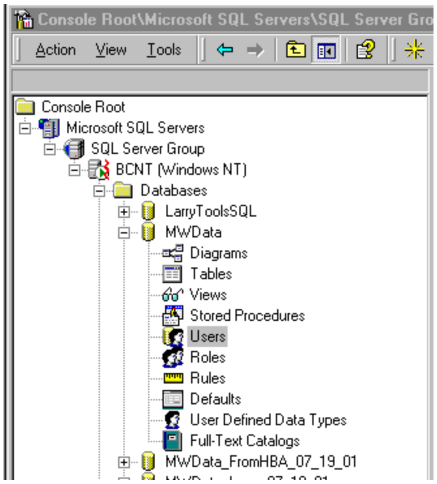
1) Do you need Roles?
If you are working on the same NT/2000 server for development, and live data, you don't need roles. All you need are your NT Server's Groups, that we added earlier.

If you are working on one NT/2000 server for development and a 2nd server for live data, you need Roles for your Groups.

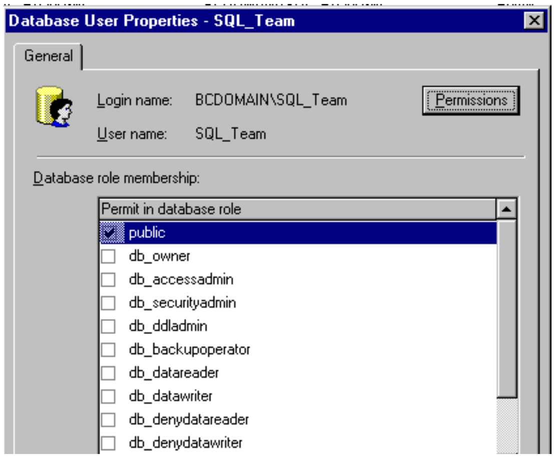
Why?
If you are using a Group SQL_Administrator on one NT/2000 server and a Group SQL_Admin_MyCustomer on another server, you will never see both groups in security in SQL Server with the exception of Roles. (*By the way it's the same situation if both groups are named the same.*) If you use Roles you will see both groups in the Roles (If you set them up right) and then be able to sent permission for each object for each role.

Maybe a better statement, an object's permission is set per role, and any group (It can be a user also, but not recommended) that belongs to the role, has the permission of that Role for that object. Any User that is a member of the Group, has the permission on the object.

If you don't use Roles and set permission based on Groups, then go to the database, Users, and double click on your group. Click the Permissions button and follow the same instructions for Roles.



- Console Root
  - Microsoft SQL Servers
    - SQL Server Group
      - BCNT (Windows NT)
        - Databases
          - LarryToolsSQL
          - MWData
            - Diagrams
            - Tables
            - Views
            - Stored Procedures
            - Users
            - Roles
            - Rules
            - Defaults
            - User Defined Data Types
            - Full-Text Catalogs
          - MWData_FromHBA_07_19_01



Login name: BCDOMAIN\SQL_Team
User name: SQL_Team

Database role membership:

Permit in database role
☑ public
☐ db_owner
☐ db_accessadmin
☐ db_securityadmin
☐ db_ddladmin
☐ db_backupoperator
☐ db_datareader
☐ db_datawriter
☐ db_denydatareader
☐ db_denydatawriter

<u>How to Use Roles.</u>

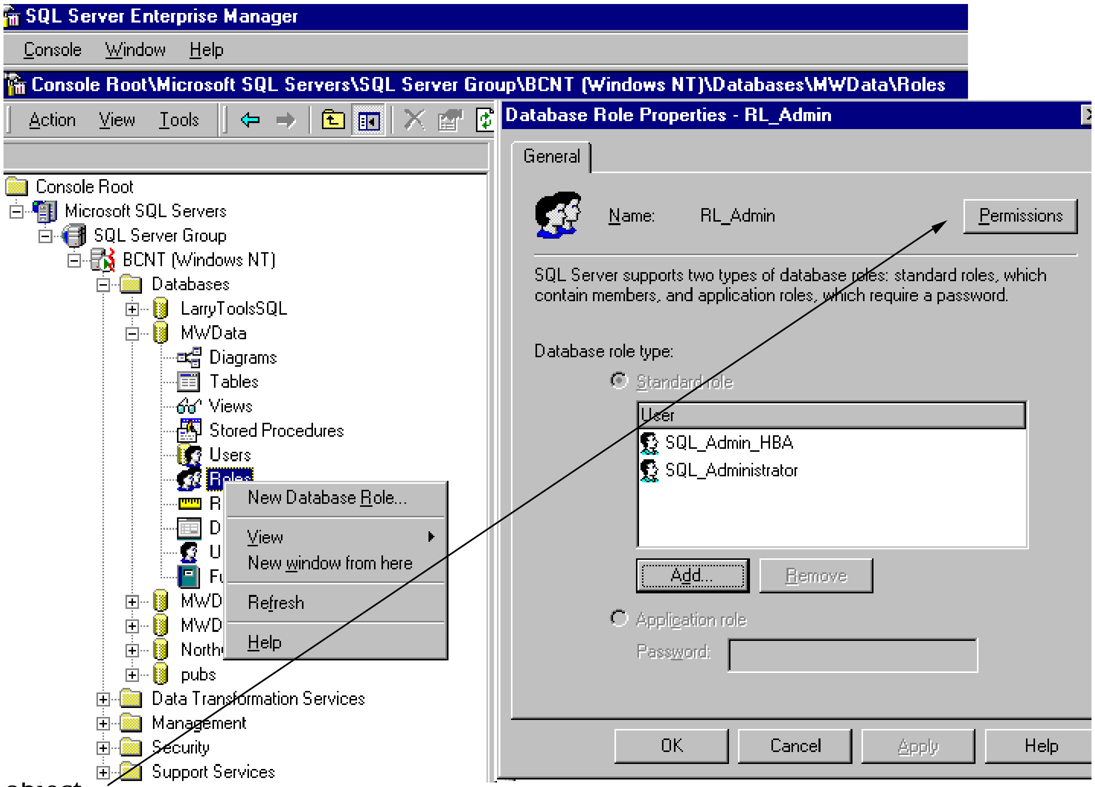
Continued next page

How to Use Roles. (Remember Roles are per database and not per server)

3) Go to your database inside SQL Enterprise Manager. Right click on Roles. Pick "New Database Roles", and create 3 database roles (for the sake of this article RL_Admin, RL_Team, and RL_ReadOnly). Each new role you add, add the approiate NT Group to the role. (eg SQL_Administrator to RL_Admin.). Note: You can add NT Groups, NT Users, SQL Server Users, and other Roles.

4) Next, press the "Permissions" button on the appropriate SQL Server Role. Go to each object, and put a check mark on the appropriate permission you want to give your user.

5) Permissions for Objects you grant from the Permission Button. If you create a new object, you must give permission to the role for that object.



SQL Server Enterprise Manager

Console Window Help

Console Root\Microsoft SQL Servers\SQL Server Group\BCNT (Windows NT)\Databases\MWData\Roles

Action View Tools

- Console Root
  - Microsoft SQL Servers
    - SQL Server Group
      - BCNT (Windows NT)
        - Databases
          - LarryToolsSQL
          - MWData
            - Diagrams
            - Tables
            - Views
            - Stored Procedures
            - Users
            - Roles
            - R
            - D
            - U
            - Fu
          - MWD
          - MWD
          - North
          - pubs
        - Data Transformation Services
        - Management
        - Security
        - Support Services

New Database Role...

View

New window from here

Refresh

Help

Database Role Properties - RL_Admin

General

Name: RL_Admin [Permissions]

SQL Server supports two types of database roles: standard roles, which contain members, and application roles, which require a password.

Database role type:

● Standard role

User
SQL_Admin_HBA
SQL_Administrator

[Add...] [Remove]

○ Application role

Password:

[OK] [Cancel] [Apply] [Help]

Grant Permission

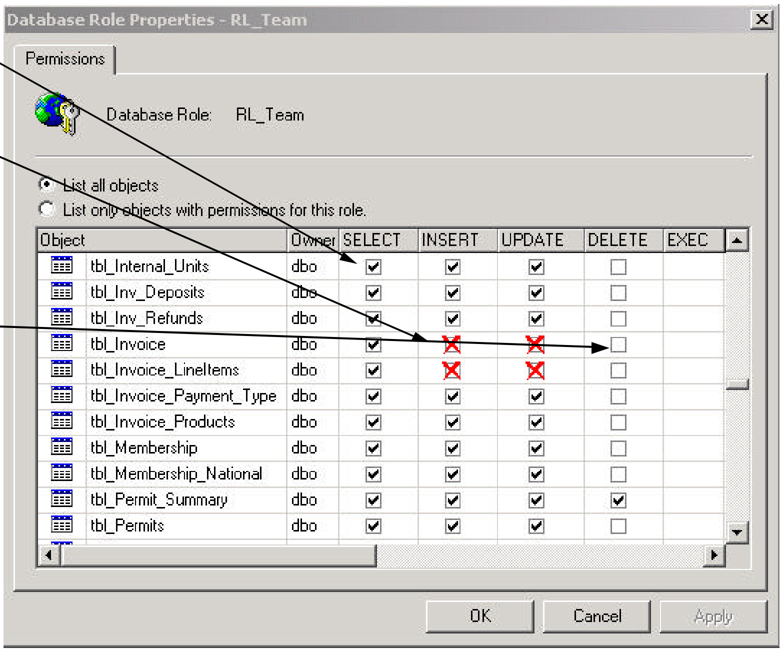
Anyone who belongs to this role has permission to Select records from this table.
Select * form tbl_Internal_Units

Deny Permission

Even if they are granted permission some where else, this says - for sure they don't have permission to Insert data in tbl_Invoice (this is not completely true, see the next section for a better explanation.)

Permission Not Granted

At this place they don't have permission to delete data, however it might be granted through some other Role (maybe SQL_AccRec role) or group or directly to the individual (I know you nor I would grant individual's rights)



Database Role Properties - RL_Team

Permissions

Database Role: RL_Team

● List all objects

○ List only objects with permissions for this role.

Object	Owner	SELECT	INSERT	UPDATE	DELETE	EXEC
tbl_Internal_Units	dbo	☑	☑	☑	□	
tbl_Inv_Deposits	dbo	☑	☑	☑	□	
tbl_Inv_Refunds	dbo	☑	☑	☑	□	
tbl_Invoice	dbo	☑	☒	☒	□	
tbl_Invoice_LineItems	dbo	☑	☒	☒	□	
tbl_Invoice_Payment_Type	dbo	☑	☑	☑	□	
tbl_Invoice_Products	dbo	☑	☑	☑	□	
tbl_Membership	dbo	☑	☑	☑	□	
tbl_Membership_National	dbo	☑	☑	☑	□	
tbl_Permit_Summary	dbo	☑	☑	☑	☑	
tbl_Permits	dbo	☑	☑	☑	□	

[OK] [Cancel] [Apply]

The above is not completely true, see Section III - Objects & Permission Subsection D for a better explanation.

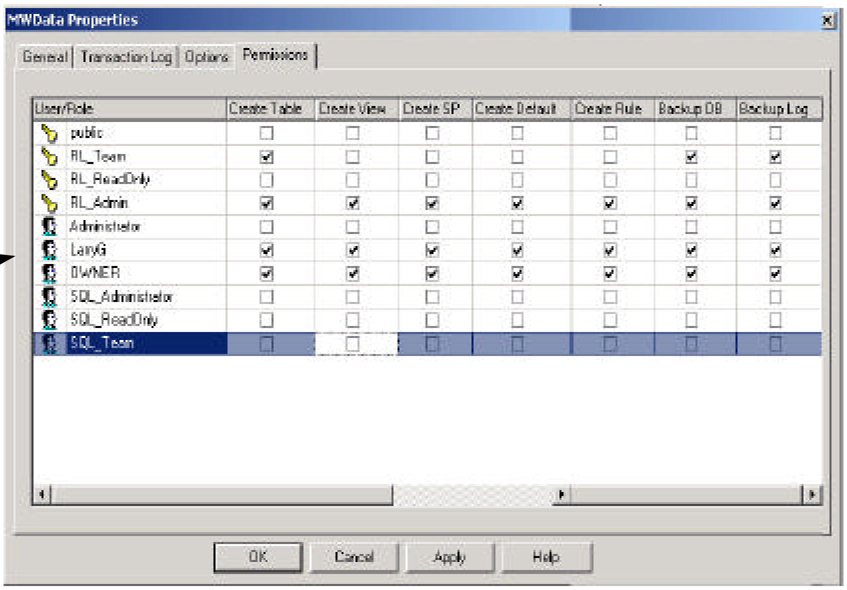
III - Objects & Permission

A) If you add an object to your database, no one has permission to use the object except System Administrators and the database owner (dbo), except the permission you give that User, Group, or Role. (This is not true if you are using the SQL Server database roles or Server roles.)

1) You need to give permission in each role that you want the people to access your object.

B) Each user has the greatest permission on an object that has been granted to the user, the Groups they belong to, or the roles they belong to.

1) SQL Server looks at all the groups and roles the user belongs to, and if any of them gives the user permission to do a process (Insert, append, delete, etc) then they have the right.
2) The exception is Denied Permission, then they do not have permission even if it is granted some where else.



User/Role	Create Table	Create View	Create SP	Create Default	Create Rule	Backup DB	Backup Log
public	☐	☐	☐	☐	☐	☐	☐
RL_Team	☑	☐	☐	☐	☐	☑	☑
RL_ReadOnly	☐	☐	☐	☐	☐	☐	☐
RL_Admin	☑	☑	☑	☑	☑	☑	☑
Administrator	☐	☐	☐	☐	☐	☐	☐
LarryG	☑	☑	☑	☑	☑	☑	☑
OWNER	☑	☑	☑	☑	☑	☑	☑
SQL_Administrator	☐	☐	☐	☐	☐	☐	☐
SQL_ReadOnly	☐	☐	☐	☐	☐	☐	☐
SQL_Team	☐	☐	☐	☐	☐	☐	☐

C) Objects and types of permission

1) Database

It is recommended that there is only one owner for all objects in a database. You can use your permission on creating objects to enforce this process.

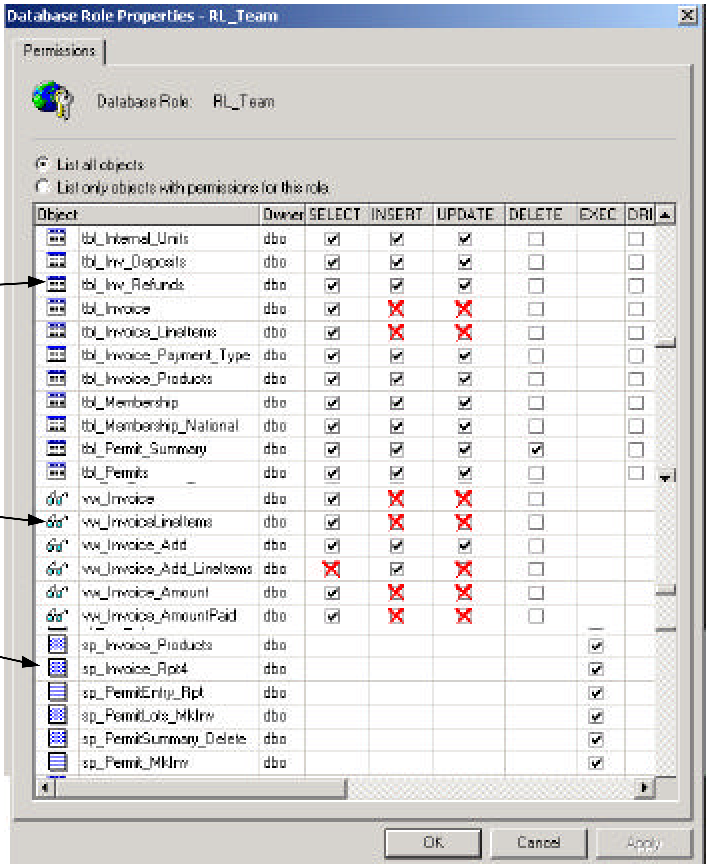
Database Properties:

per Database

2) Tables
Select - Insert - Update - Delete - DRI (Referential Integrity)

3) Views
Select - Insert - Update - Delete

4) Stored Procedures
Execute



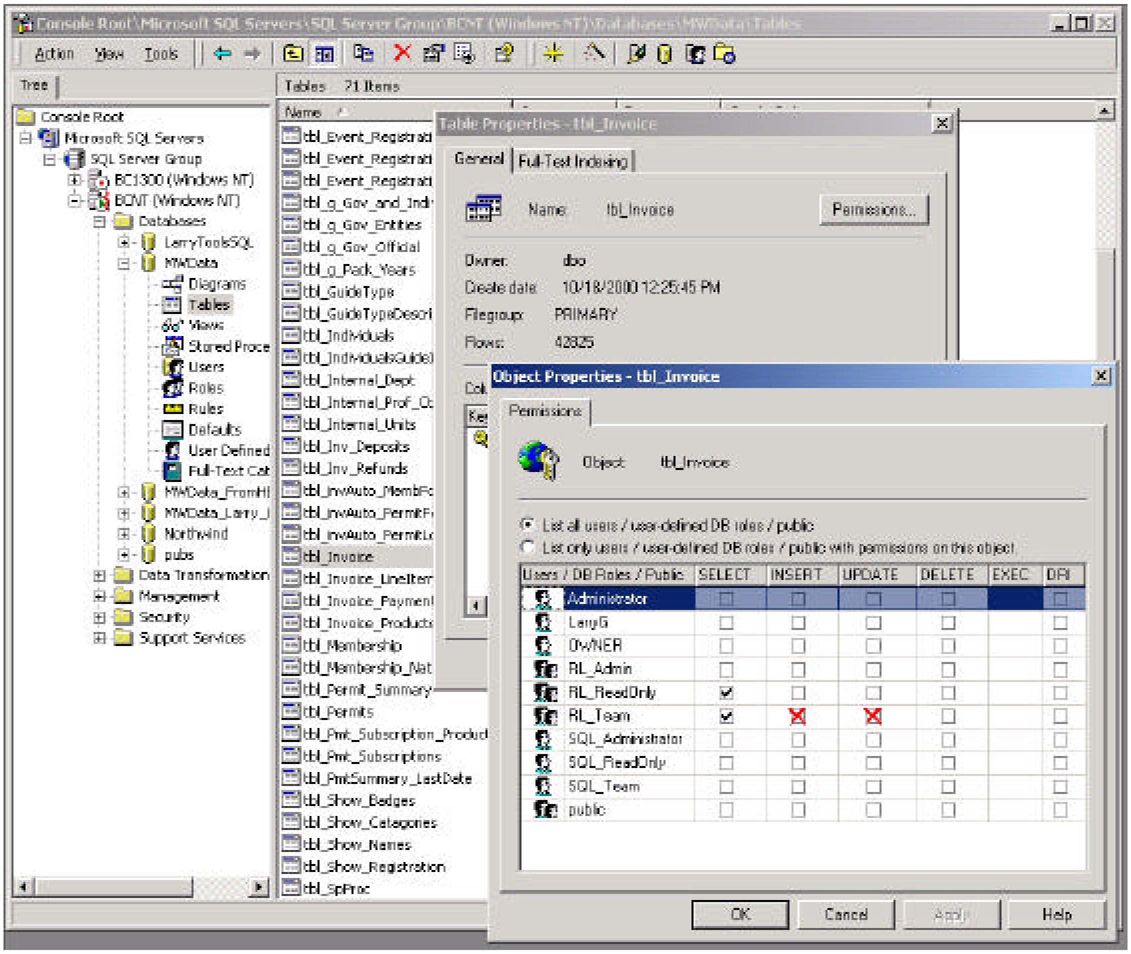
Database Role: RL_Team

● List all objects
○ List only objects with permissions for this role

Object	Owner	SELECT	INSERT	UPDATE	DELETE	EXEC	DRI
tbl_Internal_Units	dbo	☑	☑	☑	☐		☐
tbl_Inv_Deposits	dbo	☑	☑	☑	☐		☐
tbl_Inv_Refunds	dbo	☑	☑	☑	☐		☐
tbl_Invoice	dbo	☑	☒	☒	☐		☐
tbl_Invoice_LineItems	dbo	☑	☒	☒	☐		☐
tbl_Invoice_Payment_Type	dbo	☑	☑	☑	☐		☐
tbl_Invoice_Products	dbo	☑	☑	☑	☐		☐
tbl_Membership	dbo	☑	☑	☑	☐		☐
tbl_Membership_National	dbo	☑	☑	☑	☐		☐
tbl_Permit_Summary	dbo	☑	☑	☑	☑		☐
tbl_Permits	dbo	☑	☑	☑	☐		☐
vw_Invoice	dbo	☑	☒	☒	☐		
vw_InvoiceLineItems	dbo	☑	☒	☒	☐		
vw_Invoice_Add	dbo	☑	☑	☑	☐		
vw_Invoice_Add_LineItems	dbo	☒	☑	☒	☐		
vw_Invoice_Amount	dbo	☑	☒	☒	☐		
vw_Invoice_AmountPaid	dbo	☑	☒	☒	☐		
sp_Invoice_Products	dbo					☑	
sp_Invoice_Rpt4	dbo					☑	
sp_PermitEntry_Rpt	dbo					☑	
sp_PermitLots_MkInv	dbo					☑	
sp_PermitSummary_Delete	dbo					☑	
sp_Permit_MkInv	dbo					☑	

D) An object has the right granted the user for that object as long as the owner of the objects grant the permission. It makes no difference what permission the user has on the underlying tables or views.

1) If the user has no right on table tbl_Invoice (that's no permission to Select - Insert - Update or Delete), they may still have permission to enter invoices. If you give the user permission to select, Insert, Update, or Delete on the view, the fact that they have no permission on the table is not a problem. Remember you give them this permission through NT Groups they belong to, and Roles that the NT Group belongs to.
2) The same is true for stored procedures.
3) You can go to any object in the Enterprise Manager, double click to get the property form, click the permission button and see all the permission for this object.



Name: tbl_Invoice

Owner: dbo

Create date: 10/18/2000 12:25:45 PM

Filegroup: PRIMARY

Rows: 42825

Object: tbl_Invoice

● List all users / user-defined DB roles / public

○ List only users / user-defined DB roles / public with permissions on this object

Users / DB Roles / Public	SELECT	INSERT	UPDATE	DELETE	EXEC	DRI
Administrator	☐	☐	☐	☐		☐
LarryG	☐	☐	☐	☐		☐
OWNER	☐	☐	☐	☐		☐
RL_Admin	☐	☐	☐	☐		☐
RL_ReadOnly	☑	☐	☐	☐		☐
RL_Team	☑	☒	☒	☐		☐
SQL_Administrator	☐	☐	☐	☐		☐
SQL_ReadOnly	☐	☐	☐	☐		☐
SQL_Team	☐	☐	☐	☐		☐
public	☐	☐	☐	☐		☐

E) If you are programming on one SQL Server and moving to a live server you might want to look at SQL Compare from Red-Gate software (www.red-gate.com). It will not only let you move your objects from one server to another, it works well with roles and permission.