MainBoss 4.2
Installation and Administration

Revision 72

Table of Contents

MainBoss Installation and Administration Guide

Introduction

This guide provides information about installing, configuring and managing MainBoss. The first chapter gives a quick guide that should be sufficient for many users. Later chapters give more details for those with advanced needs.

This guide is aimed at providing a reasonable path to getting MainBoss up and running. The procedures and settings we describe are not the only possibilities; some sites may have good reason to take different approaches (e.g. in setting up permissions over your local area network). However, if you don't follow our suggestions and you contact MainBoss Technical Support for help, we will probably recommend that you change your configuration to match what is described in this guide.

Important Note: You will obtain MainBoss software from a page on our web site (`http://mainboss.com`). The web page will provide primary instructions for installing the software. This manual provides additional information that may be useful during the installation process, beginning with *Installation Locations for MainBoss Software* on page 4. **If there is any disagreement between the download web page and this manual, follow the instructions on the download web page.**

Software and Hardware Requirements

Operating System: MainBoss will run on any of the following operating systems:

- Windows Vista Business or better (with Service Pack 3 or later); or
- Windows Server 2008 (with Service Pack 2 or later); or
- Windows Server 2008 R2 or later; or
- Windows 7 Pro or better; or
- Windows Server 2012 R2 or later; or
- Windows 8 Pro or better
- Windows 10

For all these systems, you can use either the 32-bit or 64-bit versions.

We do not recommend Windows Vista Home, Windows Vista Basic, or Windows 7 Home.

If you will be running the **Web Access** and/or the **Web Requests** module, you must have Windows Server 2008, Windows Server 2008 R2, or better.

.NET Framework: Every computer where MainBoss will run must have the following Microsoft software installed:

□ The .NET framework, version 4.6.1 or later

Hardware: We recommend that any computer running MainBoss should have at least dual 2.5 GHz processors with 2 GB of RAM. Ideally, get the fastest RAM available. If you are running SQL Server on the same computer, add another 1 GB.

For best results, your monitor screen should have a resolution of 1280 by 1024 or better; however, it is possible to get by with 1024 by 768 if you use the “hide details panel” button:

⏬

Since MainBoss requires SQL Server, you should also check the hardware requirements of the SQL Server package you intend to use. For example, Microsoft recommends 4GB or more of memory for some versions of SQL Server.

SQL Server will run on a single computer; if you also intend to use this computer to run MainBoss, the computer’s hardware and operating system should satisfy the requirements of both SQL Server and MainBoss. Every other computer where you intend to run MainBoss has to satisfy MainBoss’s hardware and software requirements, but doesn’t have to satisfy SQL Server’s.

If you intend to run MainBoss on multiple computers, we strongly recommend that MainBoss users belong to a domain. A domain lets MainBoss users sign on with the same login name and password on different computers. You can create a domain with Microsoft Server software, or with a Linux network running Samba.

Microsoft SQL Server

MainBoss uses Microsoft’s SQL Server 2008, 2008 R2 or 2014 to read and write all data. SQL Server must be installed at your site before you can begin using MainBoss. Specifically, you must have:

□ SQL Server 2008 SP1 or SQL Server 2008 R2

□ SQL Server 2014 (any version)

□ SQL Server 2016 (any version)

□ SQL Server 2017 (any version)

Note: Microsoft no longer supports SQL Server 2005. Therefore, Thinkage no longer supports MainBoss on that version of SQL Server. If you are still using SQL Server 2005, we recommend that you upgrade to a newer version as soon as possible.

If you’ve installed MainBoss, you can determine what version of SQL Server you have by going to **Help** → **About** in the MainBoss menus.

While this Installation and Administration Guide offers suggestions of how to work with SQL Server, SQL Server is a Microsoft product and Thinkage has no control over its behavior. If you have any difficulties with SQL Server, contact Microsoft, not MainBoss support.

SQL Server is actually a family of compatible software products, with different members of the family designed for different work loads. If you have a small organization, you can use SQL Express, a free version of SQL Server available from Microsoft. For more information, see *SQL Express Considerations* on page 73.

SQL Express will likely be adequate if you intend to run the SQL Server on Windows 7 or 8 Professional. If you will have more than five simultaneous users, you should run SQL Server on Windows Server 2008 and buy a version of SQL Server whose size is appropriate to your needs.

If you intend to have multiple people using MainBoss, or you intend to access MainBoss from multiple machines, you should install SQL Server Management Studio and configuration tools at the same time that you install SQL Server. This is discussed in *Appendix A: Setting Up SQL Server* on page 70.

Your organization may already be using SQL Server for some other purpose or in connection with some other software package. In that case, you should use the same instance of SQL Server for MainBoss unless there's some good reason not to—trying to manage multiple instances of SQL Server can be confusing and prone to human error.

Microsoft Report Viewer

MainBoss prepares reports using Microsoft's Report Viewer. This version of MainBoss includes the appropriate Report Viewer software.

Note: SQL Server and Report Viewer are independent of MainBoss, including language support. In particular, reporting facilities will be in the language of your report viewer, which may or may not be the same language used by MainBoss. Similarly, diagnostic messages generated by SQL Server may or may not be in the same language as MainBoss.

Since Report Viewer is Microsoft software, Thinkage cannot take responsibility for any problems you might encounter with the Report Viewer software package.

Privileges

Windows Administrator Privileges: Normally, you do not need Windows Administrator privileges to run MainBoss. However, you do need such privileges in the following cases:

□ When you manually install MainBoss on a computer

□ When you install MainBoss Service (see *Setting Up MainBoss Service* on page 25) and when you start or stop the service

- When you set up MainBossWeb (see *Appendix C: MainBossWeb* on page 96)
- In order to look at the logs in **Administration** | **MainBoss Service** when you are first setting up the service; however, privileges are not required once the service is operating properly, since the service write event information directly into the MainBoss database

SQL Server Administrator Privileges: SQL Server Administrator privileges are needed to perform backups and restores of the MainBoss database. SQL Server Administrator privileges are also needed to add new users to the list of recognized MainBoss users (**Administration** | **Users**). This is because the new user must be given appropriate permissions to use SQL Server to access the database. However, there is a way around this; for details, see *Registering New Users with MainBoss* on page 14.

MainBoss Administration Security Role: You need the MainBoss `Administration` security role in order to use the **Administration** section of the MainBoss control panel. This security role only applies to features within MainBoss itself, and has no effect on any other software.

The person who creates a MainBoss database is automatically granted the MainBoss `Administration` security role. That person must then authorize other users, or else authorize one user and then give that user the `Administration` security role so that he/she may authorize other users.

Installation Locations for MainBoss Software

When setting up MainBoss, it’s important to distinguish between where you store the data and where you store the software:

- MainBoss *data* will be stored on a computer where SQL Server is running. There is only one copy of the data; this is shared by all users at your site.
- MainBoss *software* should be installed on every computer where people wish to work with MainBoss. You may have as many copies of the software as you like installed on any number of computers.

> The software for MainBoss doesn’t have to be installed on the server system if people won’t be using MainBoss from that system. However, a number of administrative operations are more straightforward if MainBoss *is* installed on the server system.

Problems may arise if you store the MainBoss executable program on one computer and try to run it on another computer. (This is a known problem with the Microsoft .NET framework.) As a result, you must either:

- Install the MainBoss software on every computer where you intend MainBoss to run; or
- Use ClickOnce deployment, as discussed in the next section

ClickOnce Deployment

MainBoss supports *ClickOnce Deployment*. This is a Microsoft technique which eliminates the need to install MainBoss updates on user's computers. Essentially, you create a web page that non-privileged users can use to invoke MainBoss, even if MainBoss hasn't been installed on the user's system.

Note that non-privileged users can't run MainBoss using ClickOnce unless they have prerequisite Microsoft software installed on their systems (the correct version of .NET).

For more about installing the ClickOnce version of MainBoss, see the information provided on the download web page from which you get MainBoss software.

For help in dealing with ClickOnce problems, Microsoft has a web page that discusses ClickOnce troubleshooting:

`https://msdn.microsoft.com/en-us/library/fb94w1t5.aspx`

Manual Installation

The sections that follow deal with manual installation, as opposed to ClickOnce Deployment.

Manual Installation Overview

To install and configure MainBoss, some operations must be done by someone who has Windows Administration privileges (typically someone from the IT department). Other operations must be done by someone with SQL Server Administration privileges. Finally, some operations must be done inside of MainBoss by someone with the MainBoss `Administration` security role. (In some organizations, the same person will perform all these functions.)

Below we list the recommended order of operations for installation and initial set-up, as well as the privileges needed for each step (if any).

1. Install SQL Server first. This should be on a computer that is accessible to all other computers where you wish to use MainBoss. We will call this the *Server computer*. [Windows Administration privileges]
2. Install the MainBoss software on a single computer first. Ideally, this should be the computer where SQL Server is running; this eliminates any issues that may arise when accessing SQL Server over your local area network. [Windows Administration privileges]
3. Start MainBoss on the computer where you just installed it, and create a maintenance organization database as described in *Creating a Maintenance Organization* on page 7. [SQL Server Administration privileges; anyone who creates a MainBoss database is automatically granted the MainBoss `Administration` security role on that database. The SQL Server `dbcreator` permission is *not* sufficient.]

4. If you intend to import data from MainBoss Basic (MainBoss 2.9), follow the instructions given in the guide Migration from MainBoss Basic to MainBoss. [SQL Server Administration privileges]
5. Enter the license keys you were given when you licensed MainBoss. For more information, see *Entering License Keys* on page 9. [MainBoss `Administration` security role]
6. If multiple people will be using MainBoss, you must add those people to MainBoss's table of authorized users, as described in *Users* on page 13. [MainBoss `Administration` security role]
7. Once you have MainBoss working correctly on one computer, either set up ClickOnce deployment or install the MainBoss software on all other computers where you wish to use MainBoss. [Windows Administration privileges]
8. Once the software is installed on a given computer, each person who will be using MainBoss on that computer must start MainBoss and add the MainBoss database to their personal list of known maintenance organizations. For more information, see *Installing MainBoss on Other Computers* on page 22. [Any user can add a MainBoss database to his/her list of known databases. However, a user's access to a database is controlled by the user's security roles as specified in that database.]
9. If you have licensed the **MainBoss Service** module, install MainBoss Service as described in *Setting Up MainBoss Service* on page 25. [Windows Administration and SQL Server Administration privileges]
10. If you have licensed the **Web Access** and/or **Web Request** modules, install MainBossWeb as described in *Appendix C: MainBossWeb* on page 96. [Windows Administration privileges]

The Windows operating system is full of pitfalls, especially if your IT department has set up a customized environment. For help dealing with difficulties during the installation process, see *Appendix A: Setting Up SQL Server* on page 70 and *Appendix B: Troubleshooting* on page 77, especially if you'll be using SQL Express.

Manually Installing MainBoss Software

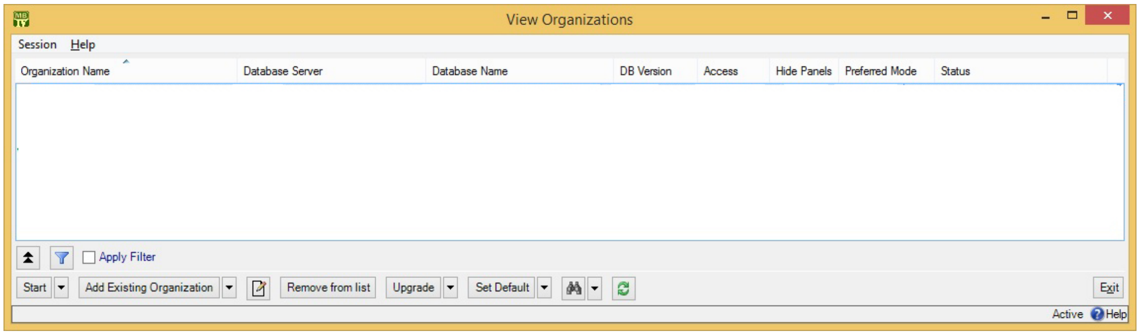
MainBoss is currently available through download from the MainBoss web site. Make sure you get the most up-to-date version of the software. To install the software, follow the instructions given on the download web page.

The instructions given on the download web page override and take precedence over instructions in this guide.

Once you've installed MainBoss on a computer, the software can be started through the standard Windows program menu. Typically, however, you should create a desktop icon that starts the program.

Creating a Maintenance Organization

When you first start MainBoss, the software displays a window that lets you create a new maintenance organization (database).

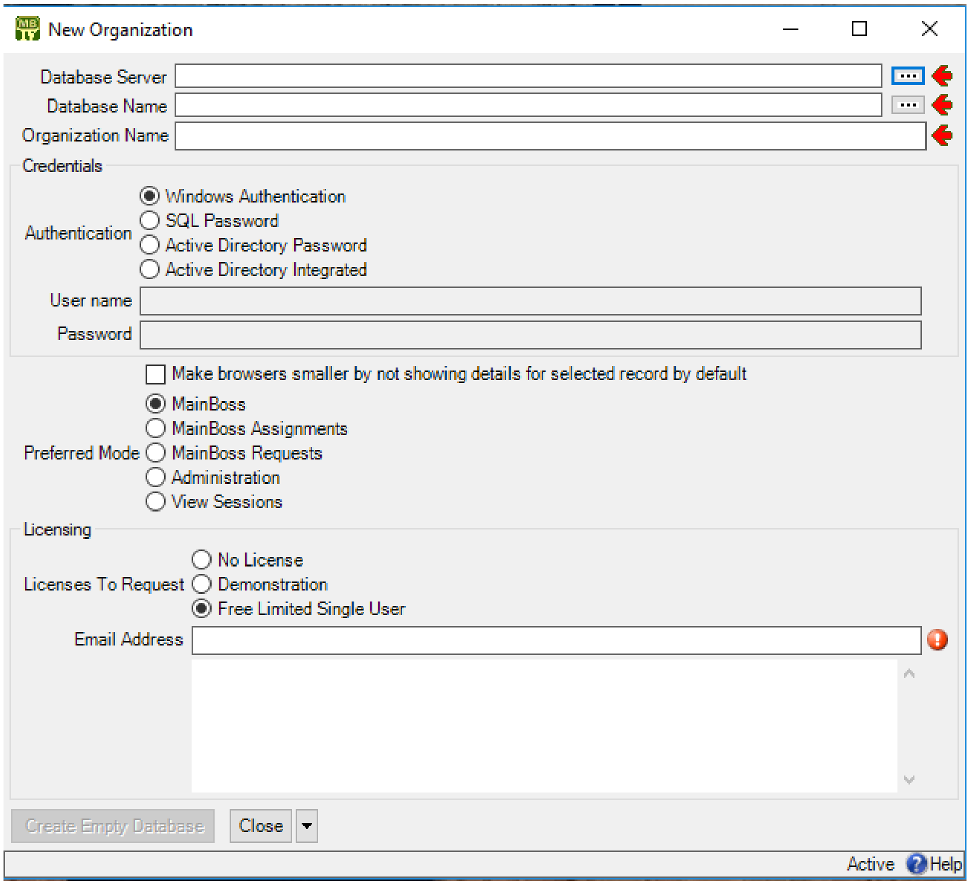


In order to create a MainBoss database, you must be logged into an account that has SQL Server Administrator privileges on the SQL Server that will manage the MainBoss database. (The SQL Server `db_creator` security role is not enough.)

Changing Regional Settings may also change the currency symbol (e.g. from “$” for dollars to “£” for pounds). However, MainBoss cannot handle multiple currencies; it assumes that all money values are in the same currency. For example, if you enter `33` in a “**Unit Cost**” field, MainBoss may display the result as `$33.00` or `£33.00` depending on your Regional Settings...but internally, MainBoss simply records `33.00`. Therefore, it is important that all MainBoss users use the same currency for all money values.

□ **To create a new maintenance organization database:**

1. Drop the arrow on **Add Existing Organization** and click **Create New Organization**. This opens a window like this:



New Organization

Database Server

Database Name

Organization Name

Credentials

Authentication ● Windows Authentication ○ SQL Password ○ Active Directory Password ○ Active Directory Integrated

User name

Password

□ Make browsers smaller by not showing details for selected record by default

Preferred Mode ● MainBoss ○ MainBoss Assignments ○ MainBoss Requests ○ Administration ○ View Sessions

Licensing

Licenses To Request ○ No License ○ Demonstration ● Free Limited Single User

Email Address

2. In “**Database Server**”, specify the name of the SQL Server that you will use.

If you click the “...” button after “**Database Server**”, MainBoss will attempt to determine what servers are available to you. However, this list may be incomplete—firewall settings and SQL Server configurations can prevent servers from appearing in the list, even though the servers may be available for use.

3. In “**Database Name**”, enter the name of the new database. This will be the name used by SQL Server; it must be different from all other databases controlled by the server. We recommend that this name should *not* contain any spaces or special characters—just letters and digits. (The default database name is `MainBoss`; this name is recommended, unless your organization has multiple MainBoss databases.)

If you click the “...” button after “**Database Name**”, MainBoss will attempt to display all databases already managed by the “**Database Server**”. You cannot choose any of these names (because the databases already exist), but the list helps you see what names are already in use. (Depending on your permissions, you may not be allowed to see the list.)

4. In “**Organization Name**”, enter the name of your company or organization. Your “**Organization Name**” can contain spaces and special characters if you wish.

5. In “**Credentials**”, select the Authentication Credentials that allow access to the SQL server and provide a User Name and Password if required.
6. Select the type of license you wish to have installed at time the database is created. Demonstration and free licenses require an email address so that the full license key can be emailed to you. You will have to enter the full license key when you receive it to continue to use the program after 30 days.
7. Click **OK**. MainBoss will proceed to create the database.

Entering License Keys

When you license MainBoss, you will be given a set of license keys for the software. Once you have installed MainBoss and created a new maintenance organization, MainBoss enters a special mode where you can enter these keys. You must have a separate set of license keys for each organization you use; for more information, contact your dealer or Thinkage Ltd.

License keys have the format

```
AAAAA-AAAAA-AAAAA-AAAAA-AAAAA
```

where each `A` is a letter or number.

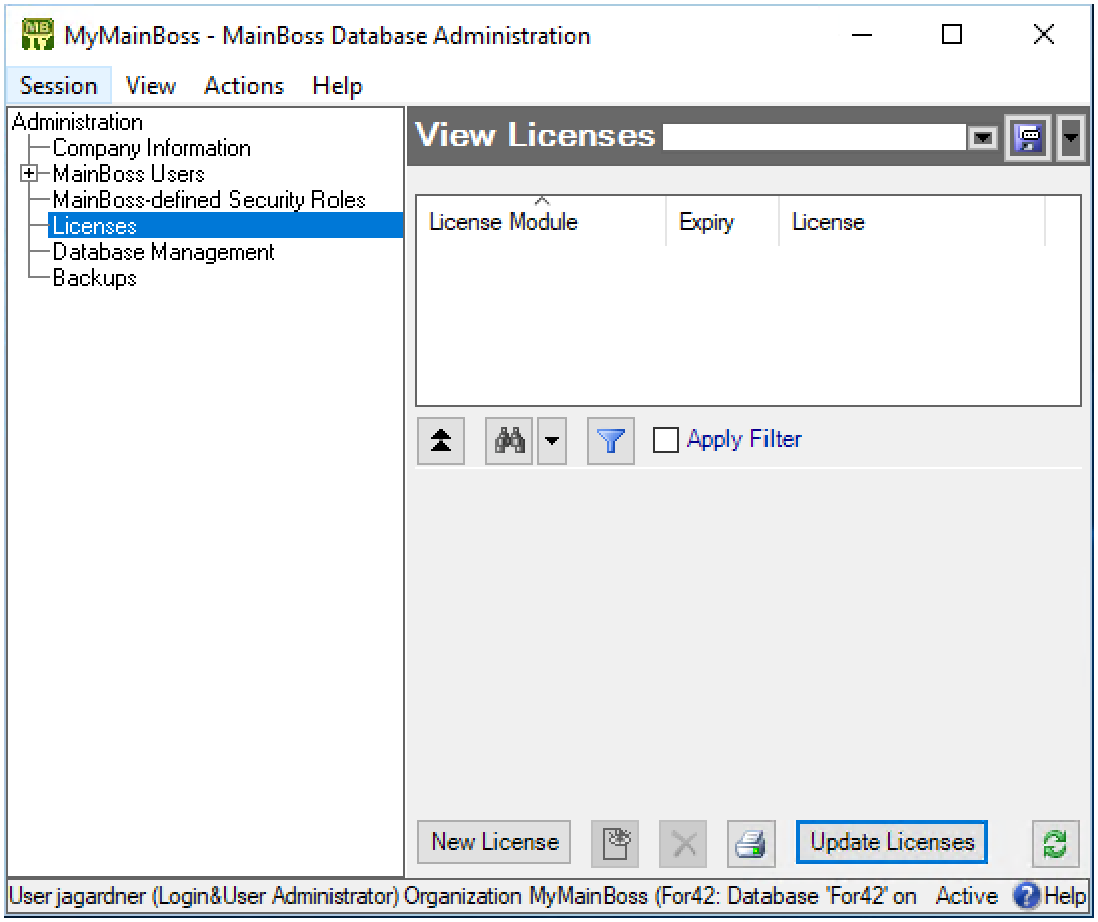
> The presence or absence of license keys will change the appearance of many MainBoss windows. For example, if you have not licensed the **Purchasing** module, you will not see functions related to purchase orders in the control panel and in other parts of MainBoss.

Thinkage gives you the option of receiving new license keys by email. This makes it possible to enter all the keys you receive by copying them from the email message and “pasting” them into MainBoss. To do so, follow these steps:

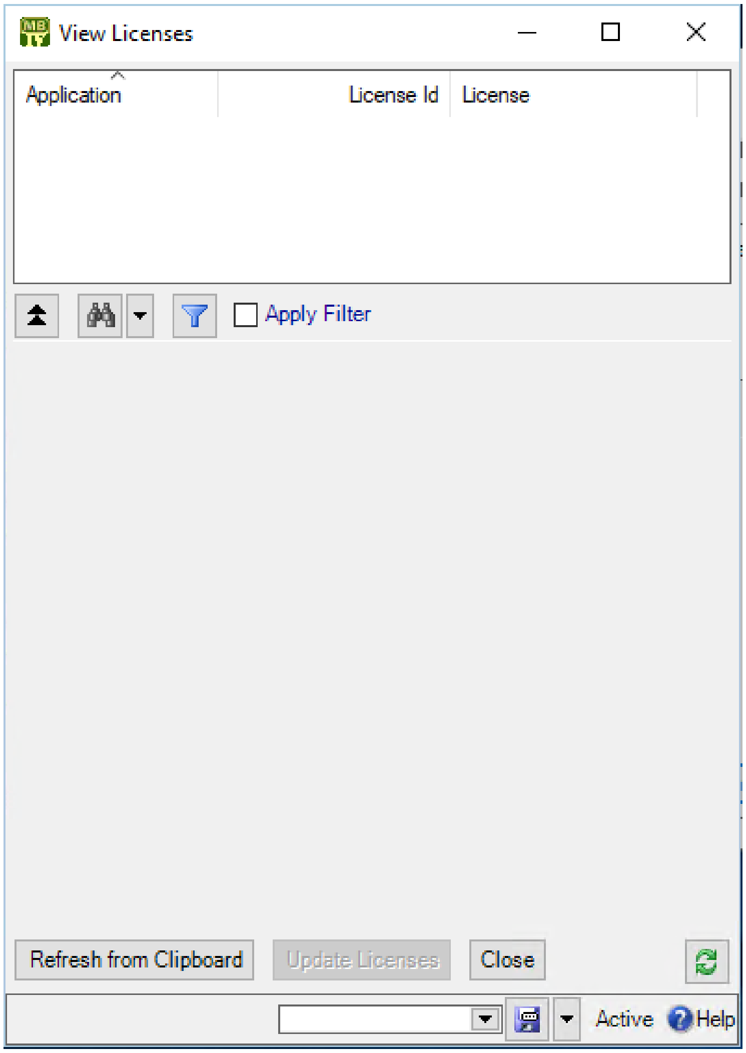
□ **To enter your license keys by copying from an email message:**

1. Open the e-mail message's PDF attachment that contains the new license keys.
2. Select the *entire* content of the attachment. You do not have to select individual keys.
3. Copy the selected content. In most email programs, you can do this by pressing **<Ctrl+C>**.

4. In the MainBoss control panel, go to **Administration** | **Licenses**:



5. Click **Update Licenses**. MainBoss opens a window where you can enter the licenses:



6. Click **Refresh from Clipboard**. MainBoss scans through the email message that you copied in Step 2, and extracts the license keys. MainBoss displays these keys in the upper part of the window.
7. Click **Update Licenses**. This uses the new license keys to replace the corresponding old ones, subject to the conditions stated below.
8. Click **Close** to close the window.

MainBoss will not replace an old key with a new key unless the new key is “better” than the old one. This means that the new key must have a later expiry date or version number than the old key.

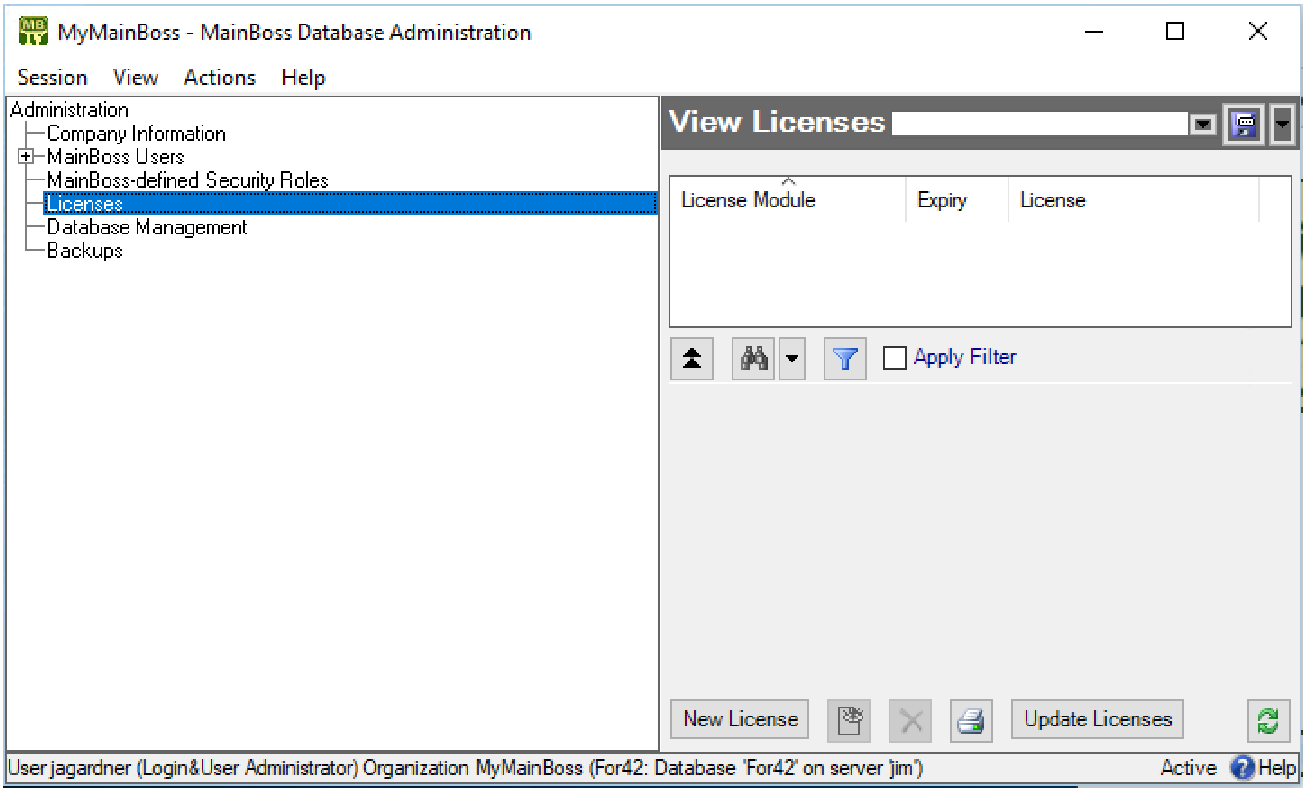
Also, all your license keys must have the same License ID as the **MainBoss** key. (If you do not have a **MainBoss** key, the other keys must have the same License ID as the **MainBoss Requests** key.)

Note: After changing keys, you must close and open the database in order for the change to take effect. This means that you must go to MainBoss's **Session** menu and select **Change Maintenance Organization**. In the resulting window, select the same organization and then click **Start**. Until you do this, MainBoss will behave as if you're still using the old license keys.

If you do not receive your license keys in a format that allows copying to the clipboard, you must enter the keys by hand:

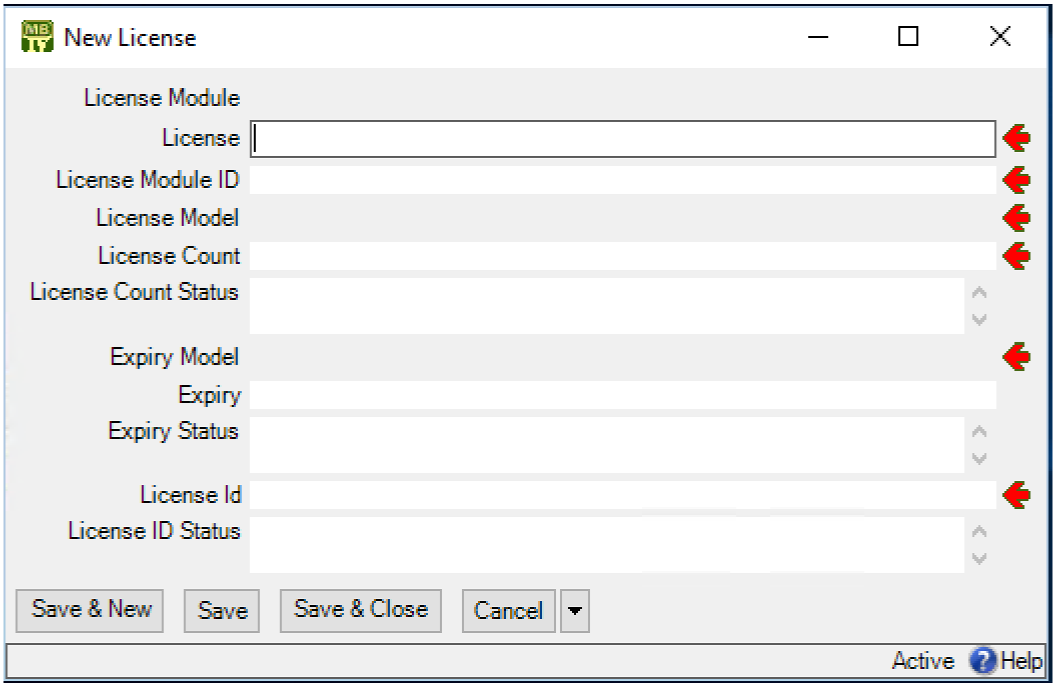
□ To enter your license keys by hand:

1. In the MainBoss control panel, go to **Administration** | **Licenses**:



User jagardner (Login&User Administrator) Organization MyMainBoss (For42: Database 'For42' on server 'jim')

2. Click **New License**. MainBoss opens a window where you can enter a license key:



3. In "**License**", enter the **MainBoss** license key (or **MainBoss Requests** if you have not purchased a **MainBoss** license). Letters can be typed in either upper or lower case—it doesn't matter which. You must type the hyphens "-" between groups of characters.

4. When you have typed the key correctly, MainBoss will fill in the rest of the window with information extracted from the key. If this doesn't happen, check the key to make sure you have typed it correctly.

5. Click **Save & New** to save what you've just typed in. MainBoss will clear all the fields and let you type in a new key.

6. Repeat Steps 3-5 until you have typed in all the keys. Click **Close** or **Save & Close** to return to the previous window. The window will show all your new keys.

Users

In order for users to access a MainBoss maintenance organization, their SQL Authentication credential names must be recorded in the MainBoss database. When a maintenance organization is first created, the only user who can access the organization is the person who originally created the database.

Important: In order for someone to be able to use MainBoss from several different computers, that person must have the same login name and password on each of the computers. (This is automatically true if you use domains.)

In the procedure described below, you record the Authentication Credentials of people authorized to use MainBoss.

The names in MainBoss's **Users** list must be individual Authentication Credential names; they can't be groups. In MainBoss, Authentication Credentials are associated with records in order to create an audit trail. This is why MainBoss itself doesn't accept group authorizations—an audit should record exactly who made a particular change, which means tracking the individual Authentication Credential, not just a group name.

Security Roles: When you create a new user record, it has no associated security roles. This means that new users have no permission to do anything in MainBoss, unless you specifically assign them roles. You must therefore assign each user at least one security role if you expect the person to use MainBoss in any way. For more on security roles, see *Security Roles* on page 19.

Note: The person who creates a MainBoss database is automatically given a set of security roles which grant full access to all MainBoss operations.

SQL Server Authorization: In order for someone to use MainBoss, the person's Authentication Credential must be registered in both the MainBoss database and with SQL Server. Only someone with SQL Server Administration privileges can register new users with SQL Server.

- The easiest scenario for adding new users to MainBoss is if someone with SQL Server Administration privileges does the work. In this case, MainBoss can automatically register each new user in the MainBoss database and with SQL Server.
- If the person who adds new users to MainBoss *doesn't* have SQL Server Administration privileges, then the process needs two separate steps:
    - Registering the new users with MainBoss, as described in *Registering New Users with MainBoss* on page 14 (which can be done by anyone who's already registered with MainBoss and SQL Server)
    - Registering the new users with SQL Server, as described in *Authorizing Users in SQL Server* on page 89 (which can only be done by someone with SQL Server Administration privileges)

Registering New Users with MainBoss

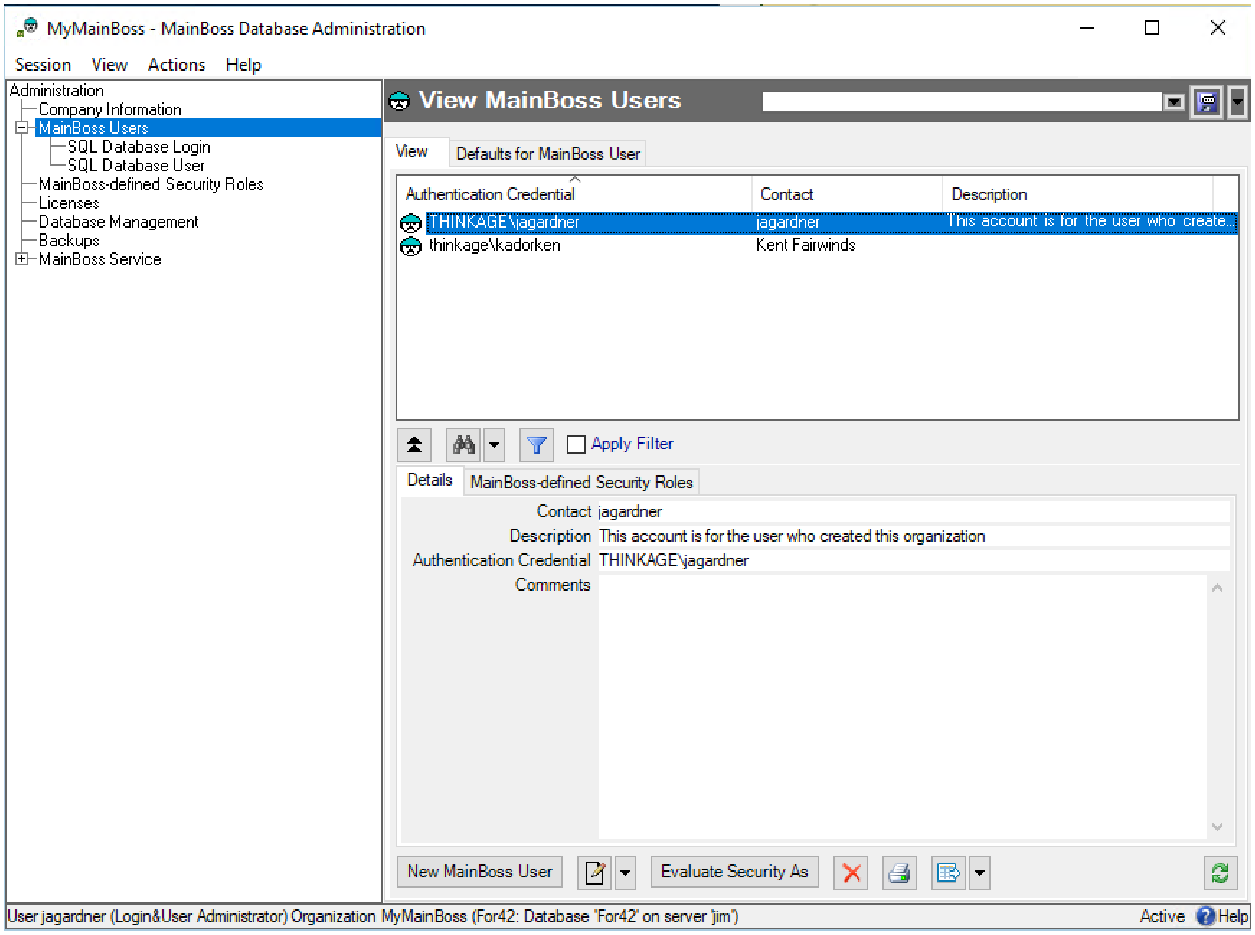
To access the MainBoss database, a user must be authorized by the instance of SQL Server that manages the database. There are two possible types of authorization:

- *Windows Authentication* means that the user's Windows login name appears in SQL Server's list of recognized login names
- Otherwise, users can identify themselves with a special SQL Server name and password. This name is separate from the person's Windows login name. Some organizations use

separate SQL Server names if several people use the same login name and/or SQL Server name. (Sharing names between multiple people is a potential security problem and is not recommended.)

When you authorize a new user to access the MainBoss database, MainBoss attempts to set up appropriate permissions for the access. In order for this to happen, you must have administration privileges on the instance of SQL Server that manages the MainBoss database. Start MainBoss, then follow these steps:

1. In the MainBoss control panel, go to **Administration** | **Users**:

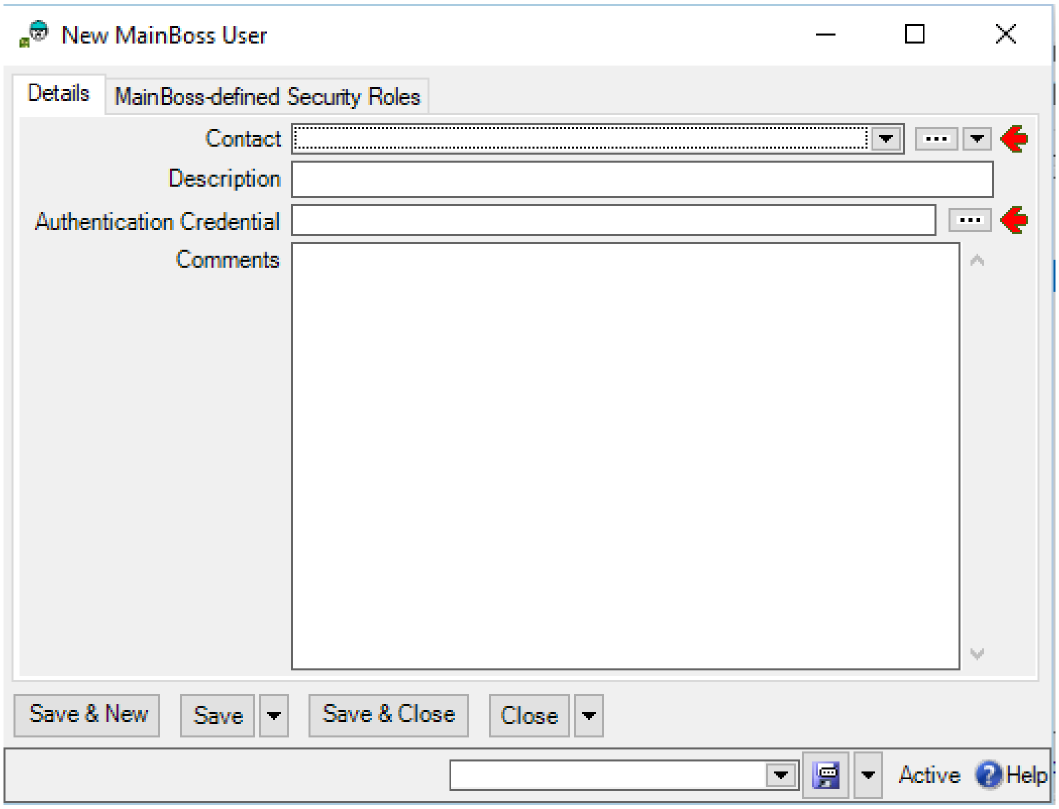


Authentication Credential	Contact	Description
THINKAGE\jagardner	jagardner	This account is for the user who create...
thinkage\kadorken	Kent Fairwinds	

Contact jagardner
Description This account is for the user who created this organization
Authentication Credential THINKAGE\jagardner
Comments

User jagardner (Login&User Administrator) Organization MyMainBoss (For42: Database 'For42' on server 'jim')

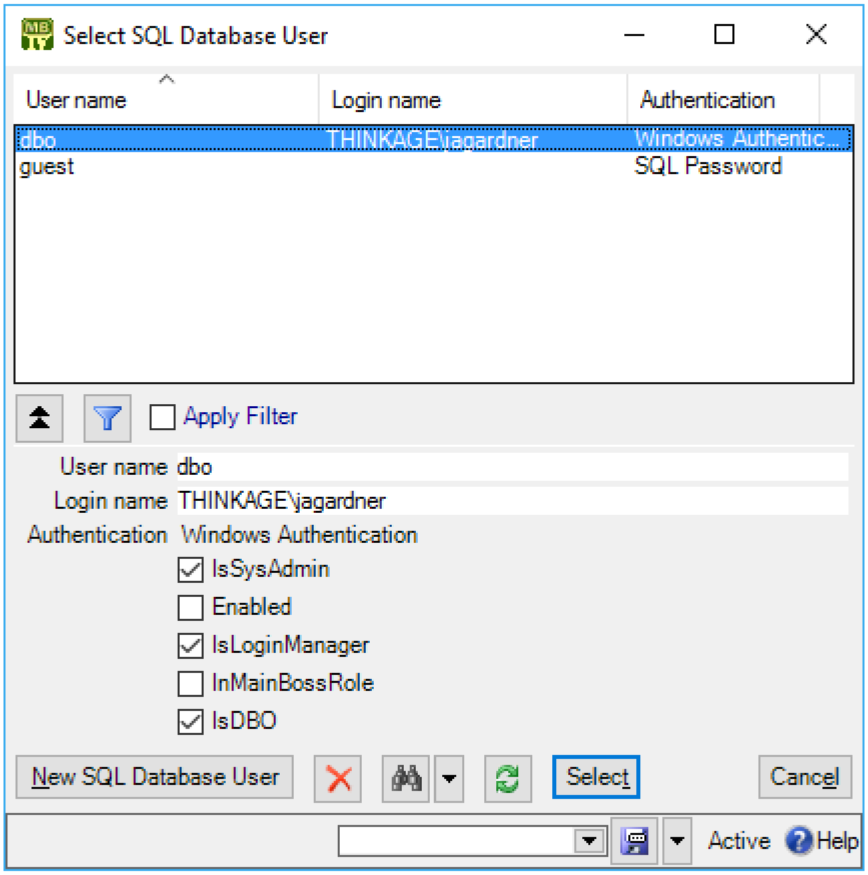
2. Click **New MainBoss User**. This opens a window where you can specify other users who can use the database.



3. On the “**Contact**” line, click the second drop-down arrow. From the resulting menu, click **New Contact**.

4. Fill in contact information for the user whose record you are creating. When you’re done, click **Save & Close** to save the contact information and close the window.

5. Back in the user record, click the “...” button associated with “**Authentication Credential**”. This opens a window that displays the people currently authorized to access databases in the instance of SQL Server that manages the MainBoss database:



User name	Login name	Authentication
dbo	THINKAGE\jagardner	Windows Authentic...
guest		SQL Password

User name dbo

Login name THINKAGE\jagardner

Authentication Windows Authentication

☑ IsSysAdmin

□ Enabled

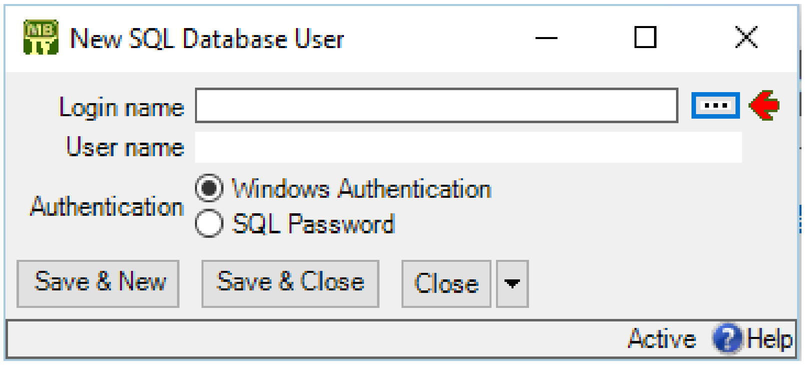
☑ IsLoginManager

□ InMainBossRole

☑ IsDBO

6. If you user you want to authorize already has an entry in this list, click the user's name and then click **Select**. You can then go on to Step 14 below.

   Otherwise, click **New SQL Database User**. This opens a window of the following form:



Login name

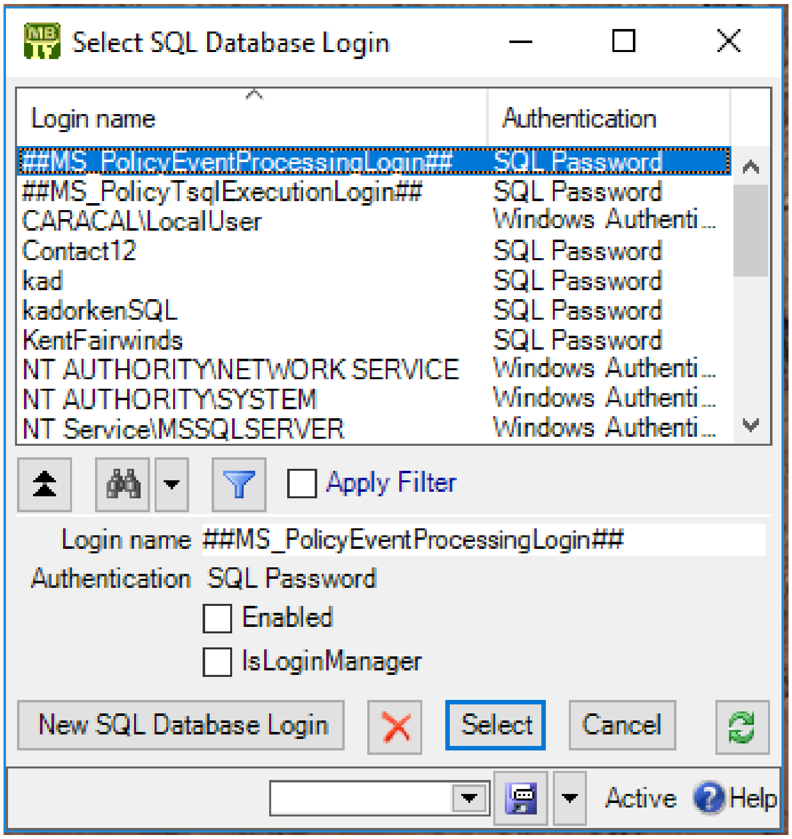
User name

Authentication ● Windows Authentication ○ SQL Password

7. If the person will access the database under a normal Windows login name, click **Windows Authentication**, then click the "..." button after "**Login name**".

   If the user will access the database under a special SQL Server name and password, click **SQL Password** and enter the password in "**Password**". Then click the "..." button after "**Login name**".

In either case, MainBoss will open a window where you can specify the login name:



Login name	Authentication
##MS_PolicyEventProcessingLogin##	SQL Password
##MS_PolicyTsqlExecutionLogin##	SQL Password
CARACAL\LocalUser	Windows Authenti...
Contact12	SQL Password
kad	SQL Password
kadorkenSQL	SQL Password
KentFairwinds	SQL Password
NT AUTHORITY\NETWORK SERVICE	Windows Authenti...
NT AUTHORITY\SYSTEM	Windows Authenti...
NT Service\MSSQLSERVER	Windows Authenti...

Login name ##MS_PolicyEventProcessingLogin##

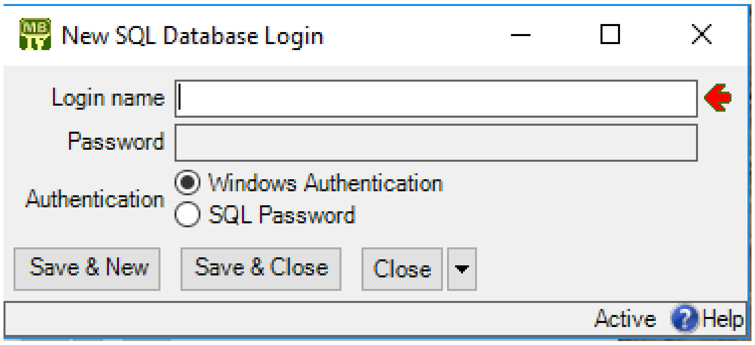
Authentication SQL Password

□ Enabled

□ IsLoginManager

8. If the user’s login name appears in the list, click the name, then click **Select**. You will return to the previous window. Fill in any information that still needs to be entered, then click **Save & Close**. Continue with Step 12 below.

   Otherwise, you have to create a new login entry for the person. Click **New SQL Database Login**. This opens a window with the following form:



Login name

Password

Authentication ● Windows Authentication
○ SQL Password

9. If the new user will be identified with a normal Windows login name, click **Windows Authentication** and enter the name in “**Login Name**”. Otherwise, click **SQL Password**, and enter a suitable name and password into “**Login name**” and “**Password**”. (For **SQL Password**, the login name must be different from other names in the SQL Server list of

logins. The user will be asked for the name and password whenever they start using MainBoss.)

10. Click **Save & Close** to return to the previous window.
11. Click the Refresh operation to get the new entry to appear, then select it and click **Select** to select the new entry in the list of logins.
12. The SQL Database User name will match the Login name. Click **Save & Close** to save the new SQL Database user record.
13. Click the Refresh operation to get the new SQL Database User to appear in the list. Select and Click a name in the list of possible SQL users, then click **Select**.
14. Back in the user record, click **Save & New**.
15. Repeat Steps 3-7 for each person who'll be using MainBoss. Click **Close** or **Save & Close** to return to the previous window. The window will show the users you have authorized.

Your MainBoss license restricts how many entries there may be in the **Users** table.

In order to add new users within MainBoss, you must have the MainBoss `Administration` security role, or else you must have Windows Administrator privileges and start MainBoss with "Run as administrator".

If you delete a user from the **Users** table and later wish to restore the same user, just double-click the word `Active` in the bottom right-hand corner of **Administration** | **Users**. This will show all the deleted users as well as the active ones. Click on the entry for the user you wish to restore, then click the drop-down button beside **View** and click **Restore**.

Security Roles

In order to use MainBoss, users need to be assigned one or more security roles. The available roles are listed in **Administration** | **Security Role**. The comments provided in each role record indicate what permissions that role provides.

Security roles affect what a user sees: if a user doesn't have permission to see a particular type of information, that information will be blanked out or missing entirely from MainBoss windows. Security roles also affect what a user can do; for example, if a user has permission to view work orders (`WorkOrderView`) but not to close them, the **Close Work Order** button will be disabled for that user.

Important: When information is blank because you don’t have appropriate permissions, move the cursor to the blank; MainBoss will tell you what security roles will give you permission to see that information. Similarly, if a button is disabled, move the cursor to the button; MainBoss will tell you what security roles will allow you to use that button. Note, however, that in some cases, information will disappear entirely (e.g. sections of records just won’t be displayed). In this case, you may not know that something is missing.

To assign security roles to a user, you use the **Security Role** section of a user record (**Administration** | **Users**). In order to decide what roles an individual user should be granted, you must consider what that user is expected to do in his/her job. For example, consider someone working on a help-desk and receiving problem reports from clients outside the maintenance department. What does that person need to do?

- He or she will need to create work requests describing problems; that requires `Request` permission.
- Help-desk personnel also need a small amount of information on work orders (so that they can tell a client, “Your work has been scheduled for Thursday”). This suggests the `WorkOrderSummary` security role.
- The `UnitSummary` role might also be useful. It gives help-desk personnel enough information about units to make sure they understand which unit the client is talking about.

Other roles may also be appropriate, depending on what help-desk personnel are expected to do, and on your organization’s policies.

The MainBoss Configuration guide offers suggestions on which security roles are appropriate for various duties within a maintenance organization.

Related Security Roles: Security roles tend to come in related groupings; for example, `WorkOrder`, `WorkOrderView`, `WorkOrderFulfillment`, `WorkOrderClose`, `WorkOrderSelfAssign` and `WorkOrderSummary` are all roles related to work orders.

In such a grouping, some roles automatically include others. For example:

- `WorkOrder` automatically includes all the other roles in the grouping.
- `WorkOrderView` automatically includes `WorkOrderSummary`.

For this reason, there’s no point in giving someone both `WorkOrder` and `WorkOrderView` (for example), because `WorkOrder` automatically includes all the permissions of `WorkOrderView`.

In general, a `View` role automatically includes the related `Summary` role and a role with a simple name automatically includes roles of the same name with extra words (e.g. `PurchaseOrder` automatically includes `PurchaseOrderView`, `PurchaseOrderSummary`, and `PurchaseOrderReceive`).

Users with SQL Server Administration Privileges: If a user with SQL Server Administration privileges starts MainBoss, MainBoss automatically gives that person the MainBoss `Administration` security role. This is intended to make it possible for a privileged user to change security roles, in the event that there's no one with the MainBoss `Administration` role. (This might happen, for example, if your current MainBoss Administrator leaves the company.)

As a side effect, you cannot "downgrade" the MainBoss security permissions of anyone with SQL Server Administration privileges. That person always has the MainBoss `Administration` security role, even if the person isn't a registered MainBoss user.

Testing Security Roles

It can be difficult to figure out the effects of various security roles, especially when a user has several different roles combined. To test what a particular user can and can't do, you can use the **Evaluate Security As** button in **Administration** | **Users**. (You must have the `Administration` security role to use **Evaluate Security As**.)

If you select a user in the **Users** table, then click **Evaluate Security As**, your MainBoss session changes to have the same security permissions as the selected user. If certain types of data are invisible to that user, they'll become invisible to you too; if certain buttons are disabled, they'll be disabled for you too.

In this way, **Evaluate Security As** lets you see all the consequences of a user's security roles. Once you've examined what the user can and can't do, you can return to your own security roles by going to the **Session** menu and clicking **Reset to user YOURNAME security**.

Note: If you use **Evaluate Security As** to evaluate the security roles of someone who doesn't have `Administration` or `AdministrationView`, the first thing that will happen is that you lose permission to see **Administration** | **Users** and you may receive an error message. This may look as if you're being told that you don't have permission to use **Evaluate Security As**; however, it really means that the person you're evaluating doesn't have that permission.

Evaluate Security As gives you the same security roles as the selected user, but it does *not* change who you are. In particular, you keep your own set of assigned requests, work orders and purchase orders. It also doesn't give you SQL Server permissions if you don't already have them. This can lead to unusual error messages if you try to do certain operations. Since you're partly one user and partly another, we recommend that you avoid doing significant work while using **Evaluate Security As**; it can lead to confusion.

If you use **Evaluate Security As** to "become" another user, unusual results may occur if you already have other MainBoss windows open. For example, suppose you have a window open for editing work orders and then you use **Evaluate Security As** to become a user who has no access to work orders. If you try to use the other window to edit the work order, you may receive odd error messages—you are currently supposed to be a user with no access to work orders, so MainBoss is confused about how you can have a work order window open. While this probably won't cause any lasting difficulties, we recommend that you close all other windows before you start using **Evaluate Security As**.

Roles for Administering MainBoss

If you are administering MainBoss, you should have the MainBoss `Administration` security role. You are automatically granted this role (and others) if you're the one who creates the MainBoss database.

The `Administration` role gives you permission to assign security roles to any user, including yourself.

Installing MainBoss on Other Computers

Once you have set up MainBoss on one computer, and are sure that it is working correctly with SQL Server, you can set up MainBoss on other computers (if you wish).

A MainBoss license dictates the number of people who may be authorized to use MainBoss. More specifically, it dictates the number of users who may be entered in the **Users** table (**Administration** | **Users**). Your license does *not* affect the number of computers where the software may be installed; therefore, you may install MainBoss on any number of computers within your organization.

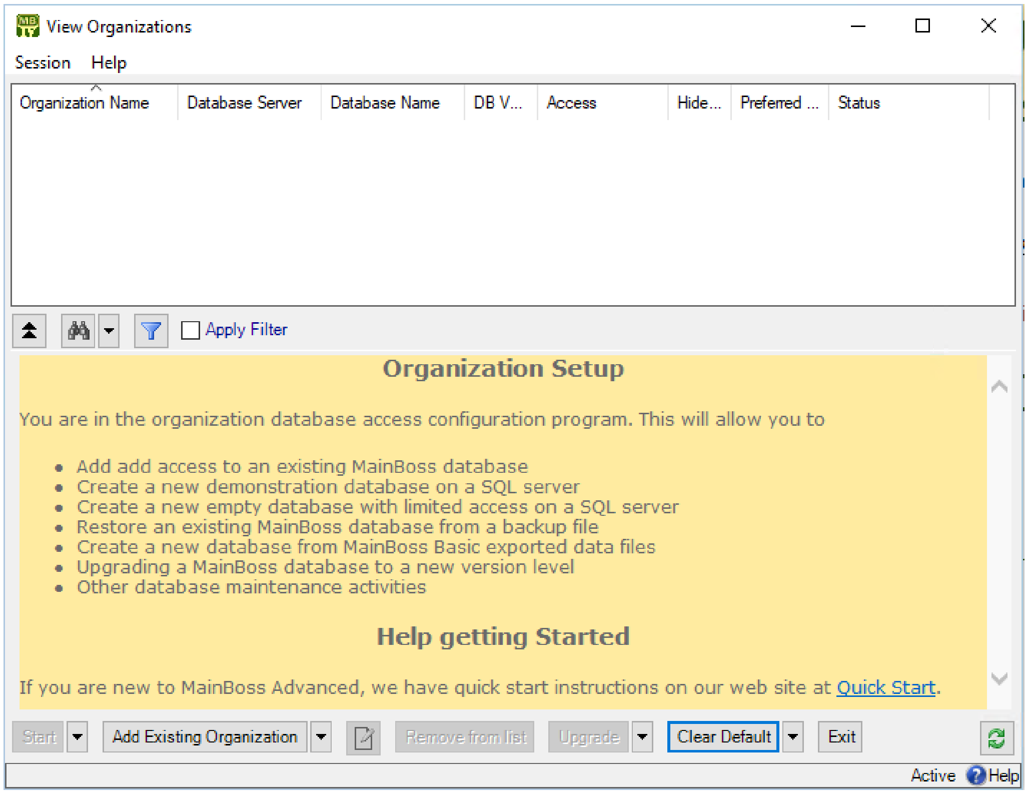
`

To install MainBoss manually on any computer, follow the steps given on the web page where you obtained the MainBoss software. Once you've done so, **each person who uses MainBoss on that computer** must set up his or her options for connecting with the MainBoss database. The alternative is to use ClickOnce deployment, as described in *ClickOnce Deployment* on page 5.

Important: MainBoss options are set up for each individual user. This means that each user must login and set up a connection to the MainBoss database. The process is easy, but the MainBoss Administrator must tell every user the "**Database Server**" and "**Database Name**" needed for the connection. Alternatively, you can have each user log in and then the administrator takes over to set up the database connection.

□ **To set up MainBoss so that it can access an existing maintenance organization:**

1. A MainBoss user should login to the computer under his/her usual login name.
2. Start MainBoss on the new machine. It will display a window asking you to specify the maintenance organization:



Organization Setup

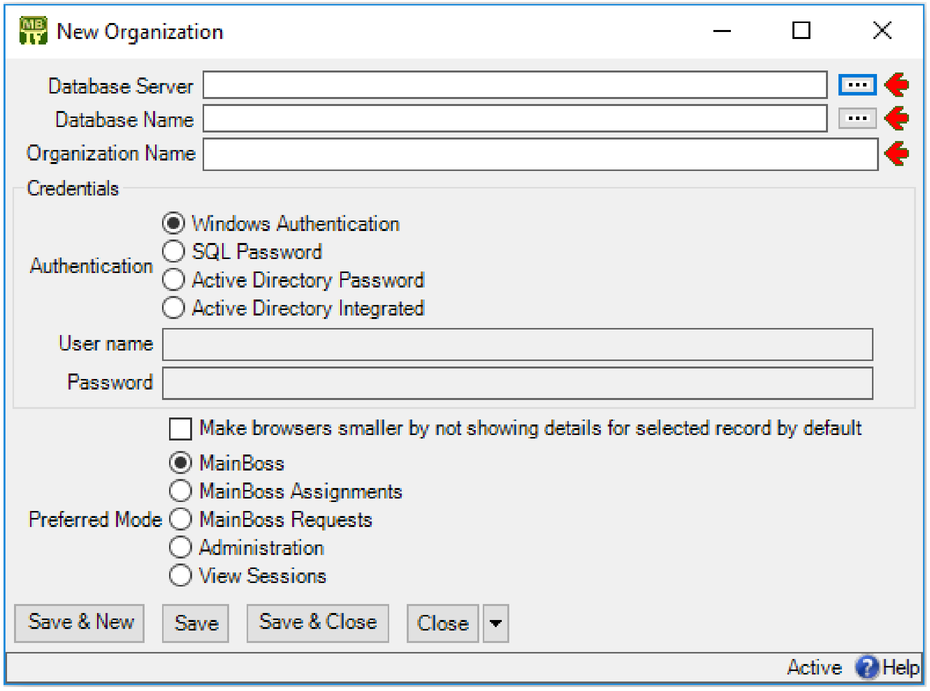
You are in the organization database access configuration program. This will allow you to

- Add add access to an existing MainBoss database
- Create a new demonstration database on a SQL server
- Create a new empty database with limited access on a SQL server
- Restore an existing MainBoss database from a backup file
- Create a new database from MainBoss Basic exported data files
- Upgrading a MainBoss database to a new version level
- Other database maintenance activities

Help getting Started

If you are new to MainBoss Advanced, we have quick start instructions on our web site at Quick Start.

3. Click **Add Existing Organization**. This opens a window like this:



Database Server
Database Name
Organization Name

Credentials

Authentication ● Windows Authentication ○ SQL Password ○ Active Directory Password ○ Active Directory Integrated

User name

Password

□ Make browsers smaller by not showing details for selected record by default

Preferred Mode ● MainBoss ○ MainBoss Assignments ○ MainBoss Requests ○ Administration ○ View Sessions

4. Set "**Database Server**" and "**Database Name**" to the same values specified when the database was created—see *Creating a Maintenance Organization* on page 7. The easiest

way to set these fields is to click the "..." buttons associated with each field. These display lists of appropriate values. However, under some conditions (related to firewall settings), the "..." lists may not supply the values you want; in that case, you must type in the values by hand.

5. If you specify "**Database Name**" using the "..." button, MainBoss will fill in "**Organization Name**" with the company name specified in the database (from **Administration** | **Company Information**). If you wish, you can change "**Organization Name**" to your own private name for the organization; this will not affect any other users. (You might change the "**Organization Name**" if you have a number of organizations you must keep track of, and you want to make sure their names are distinctive.)
6. Choose the Authentication method and provide credentials if required. The default Windows Authentication requires no User name or password.
7. Click **OK**. MainBoss will add the database name to the list.
8. Click **Start** to start MainBoss using the selected database.

Once you have set up this computer to access the maintenance organization, you don't have to do it again.

MainBoss Service

This section only applies if you have licensed the **MainBoss Service** module.

MainBoss Service handles communications between MainBoss and other applications. This includes sending and receiving all email messages to and from requestors and assignees, as well as interacting with your web site if you have the **Web Access** and/or **Web Requests** module.

Note: In previous versions of MainBoss, MainBoss Service was obtained as its own separate download. Now, it is downloaded as part of the standard MainBoss package. The process of set-up and configuration has also changed substantially.

General Principles of MainBoss Service

In order to use MainBoss Service, you must activate a Windows service. This service watches for incoming email sent to the address associated with MainBoss Service. The service also sends out acknowledgement and notification messages when appropriate.

MainBoss Service can only be activated on a single computer; MainBoss won't let you install it on more than one computer. You need Windows Administrator privileges to install MainBoss Service.

Important: MainBoss Service is designed to run continuously. It should not be installed on a computer that is regularly turned off. (If the computer *is* turned off, information will not be lost. However, incoming and outgoing messages won't be processed until the computer is turned back on.)

You will find it easier to work with MainBoss Service if you install MainBoss itself on the computer where MainBoss Service will run. However, this isn't absolutely necessary.

If you stop MainBoss Service, incoming messages will stop being processed. When you restart the service, it should begin processing the backlog of messages that built up while the service was off. The same principle applies if you shut down the computer where MainBoss Service is running.

Exotic Email Addresses: The Internet's "official" rules for defining email addresses allow for a great range of constructions that are almost never actually used. For example, parentheses can be used inside addresses to enclose comments as in

```
jsmith(comment)@site.com
```

In addition, the part before the "@" character is supposed to be regarded as case-sensitive, so that `jsmith` and `JSmith` are two different email addresses.

However, MainBoss does not distinguish between letter cases in any part of the email address, nor does it allow exotic constructions like parenthesized comments. MainBoss follows the same conventions as Windows: email addresses are treated as case-insensitive, and the only characters allowed are letters, digits, underscores, dots and hyphens.

(Note that MainBoss will still receive email messages sent from exotic addresses. However, MainBoss won't send mail to such addresses or record such addresses in fields where "normal" addresses are expected.)

Setting Up MainBoss Service

The general process for setting up MainBoss Service is described below. **You can configure and test MainBoss Service without any special Windows privileges**; however, when you're finished testing and are ready to install the service for real, you must have access to a login name with Windows Administration privileges and SQL Server Administration privileges.

When you first start setting up your MainBoss Service configuration, you can test it on any computer where MainBoss is installed. However, you will likely find it easier to do all your testing on the computer where MainBoss Service will eventually run. You should definitely test your configuration on that computer before actually installing MainBoss Service as a Windows service.

Some of the steps listed below may take a significant length of time. During this time, all buttons in the MainBoss window will be disabled.

1. In MainBoss, go to **Administration** | **MainBoss Service** | **Configuration** and click **New MainBoss Service Configuration**.
2. Enter configuration information for MainBoss Service, as described in *Configuring MainBoss Service* on page 27. You will need the MainBoss `Administration` security role, but will not need other privileges.
3. Once you have configured MainBoss Service, you are ready to test your configuration. On the computer where you intend to run MainBoss Service, login under a login name that has Windows Administrator privileges.
4. Start MainBoss on that computer.
5. To test your configuration, go to **Administration** | **MainBoss Service** and click **Manually run the MainBoss Service** on the second drop-down button at the bottom of the window. (This can be done on any computer, but only if MainBoss Service is not currently running.)

   MainBoss will start MainBoss Service and check the configuration for valid information. If any requests have already been received by email, MainBoss Service attempts to process them. MainBoss Service also attempts to send out any required acknowledgement messages. When this processing has finished, MainBoss Service quits. (**Manually run the MainBoss Service** does *not* install MainBoss Service; it simply does a test-run of the software.)

> When you test MainBoss Service in this manner, the software runs under your own login name. When you run MainBoss Service as a service, it will typically run under the login name `Network Service`. This may make a difference between test runs and real operations.

6. If MainBoss Service encountered any problems during the previous steps, they will be displayed on the screen and in the log of **Administration** | **MainBoss Service**. If any messages appear, they indicate problems with your MainBoss Service configuration. You can get additional problem diagnostics by clicking **MainBoss Service Configuration Diagnostics** on the second drop-down button at the bottom of the window. Correct the problems, then test them again as shown above.

> The drop-down button labeled **Delete Trace Logging Entries** provides a number of commands that let you control the diagnostic messages issued by MainBoss Service. It also lets you clean out old entries that are no longer relevant. By using these actions, you can simplify the configuration process by increasing or decreasing the quantity of messages that MainBoss Service produces.

7. When you can test-run MainBoss Service without configuration errors, you are ready to set up the service to run automatically. Go to the computer where MainBoss Service will run, and login under a login name with Windows Administration privileges and SQL Server Administration privileges.
8. Start MainBoss.

9. Go to **Administration** | **MainBoss Service** and click **Configure the Windows Service for MainBoss**. MainBoss will create a MainBoss Service configured with all the information you have set up.

   Note: Depending on how your Windows system and MainBoss are set up, it’s possible that MainBoss Service won’t start correctly. If so, the MainBoss Service log files won’t contain any messages (since MainBoss Service never starts running). In this case, it may be useful to check the Windows Event Viewer to see if there are any relevant event messages for MainBoss. (If you are working on one computer and trying to run MainBoss Service on a different computer, check the event logs on both comptuers.) Also check under **Administration** | **Database Management** (in the MainBoss control panel) to see if there are any relevant messages.

10. Start the service running by clicking **Start Service** (on the **Restart Service** drop-down button). If you are working on one computer and trying to run MainBoss Service on a different computer, starting the service can take a very long time.

11. Check the message log in **Administration** | **MainBoss Service** to make sure that everything is running correctly. Also check the “**Status**” line near the top of the window to make sure it says that the service is running.

If you ever want to change your MainBoss Service configuration, follow these steps:

1. Edit the existing configuration record in **Administration** | **MainBoss Service** | **Configuration** and **Save** the changed record.

2. Go to **Administration** | **MainBoss Service** and click **Configure the Windows Service for MainBoss**. This sends a message to Windows and MainBoss Service to update the configuration. It may take some time for the process to actually happen; during this time, the buttons on the window will be disabled.

3. When the appropriate processing has finally taken place, the buttons will be enabled again. Click **Restart Service** to shut down the existing service and to start a new one with the new configuration. (You will need Windows Administration privileges for this to work.) Note that if the service is running on a different computer, it may take several minutes for this operation to be completed.

Configuring MainBoss Service

The first time you configure MainBoss Service, you create a configuration record by clicking **New MainBoss Service Configuration** in **Administration** | **MainBoss Service** | **Configuration**. Once you have created an initial configuration, you can edit the configuration by clicking the 📝 button in the same window.

You can edit the MainBoss Service configuration record on any computer where MainBoss is installed. However, you can only use **Configure the Windows Service for MainBoss** (in **Administration** | **MainBoss Service**) on the computer where MainBoss Service runs.

Permissions for Configuring MainBoss Service: Installing and starting MainBoss Service requires Windows and SQL Server Administration privileges. In many organizations, maintenance personnel will not have such privileges; at the same time, people who do have such privileges may not have much experience with using MainBoss. This can lead to difficulties in configuring the service properly.

To give maintenance personnel as much freedom as possible, MainBoss has a feature that lets you specify configuration information even if you don't have Windows Administration privileges. In particular, if you have the MainBoss `Administration` security role, you can *test* a MainBoss Service configuration even if you don't have Windows Administration privileges.

This means that you can set up a configuration, test it, and see if any diagnostic messages result. You can then change the configuration and try again, repeating as often as necessary. Once you have a configuration that doesn't result in diagnostic messages, you can get someone with Windows Administration privileges to start up the real MainBoss Service using your finished configuration.

MainBoss Service MailBox: When you configure MainBoss Service you may specify a mailbox where people can send problem reports. You may also specify how MainBoss Service can send and receive messages from that mailbox.

MainBoss can receive email messages using a variety of techniques; these are variations of the POP3 or IMAP4 protocols.

For more information on SMTP, POP3 and IMAP, see the documentation for your email software.

MainBoss can automatically determine which protocol(s) your email software supports. We recommend that you mark the **Automatically Determine** option (discussed below). If you do, MainBoss tests for secure (encrypted) techniques first and only chooses plain POP3 or IMAP4 if no other technique works. If you specify a name in "**Mailbox Name**", MainBoss doesn't try POP3 or any of the related variants, since "**Mailbox Name**" only applies to IMAP4. (If required, you can force MainBoss to use one of the POP3 variants by clicking the appropriate option button.)

When MainBoss automatically determines which technique(s) your mail software supports, it only checks the default port numbers. If your mail server uses a different port than the default, you must mark a specific connection technique (e.g. POP3S) and you must enter the port number in "**Override Default Port**".

The window for configuring MainBoss Service contains the following:

Details section: Specifies basic information about MainBoss Service for this organization.

Service Name: Will be the name of the Windows service. This must be different from the names of all other Windows services running on this computer. The name should only consist of alphanumeric characters, with no blanks or punctuation charcters.

By default, “**Service Name**” is `MainBossService`. If you have multiple MainBoss databases, each one should have its own MainBoss Service and each should have a unique service name. Once you set the name of a service, it cannot be changed.

Description: Additional information about the configuration record.

Automatically Create Requestors, **Automatically Create Requestors From Active Directory**, **Automatically Create Requestors From Email**: When MainBoss Service receives an email message, it checks the address of the message’s sender against the list of authorized requestors (**Coding Definitions** | **Requests** | **Requestors**). For each authorized requestor, MainBoss Service checks whether the sender’s email address matches the “**Email**” or “**Alternate Email**” address in the requestor’s contact record (**Coding Definitions** | **Contacts**). If the sender’s email address matches either recorded address, the sender is accepted as an authorized requestor and the request is accepted.

If the sender’s email address does *not* match any known requestor, you can configure MainBoss Service to attempt to create a new requestor record for the sender. This process is controlled by the three **Automatically Create** checkboxes. You may checkmark any or all these boxes, or leave them blank; each choice leads to different results.

If you do not checkmark any of the **Automatically Create** checkboxes, MainBoss Service never creates requestor records. If a request comes in from a non-requestor, it will be rejected. (However, the rejection may not happen immediately—see the description of “**Manual Processing Time Allowance**” below.) The only way to authorize someone to be a requestor is for a person with appropriate MainBoss security roles to create a requestor record manually.

If you checkmark any or all of the checkboxes, MainBoss Service begins by determining if there is a deleted requestor record for the message’s sender. If there is such a record, it will be restored (undeleted) and MainBoss Service will process the email request as if it was sent in from that requestor. Note that this happens no matter which **Automatically Create** checkbox was checkmarked.

If there are *multiple* deleted requestor records corresponding to the sender’s email address, MainBoss Service cannot decide between them. The email message will be marked with the error message `Senders email address matches multiple Requestors` and the message will be marked for rejection. (However, it will not be rejected immediately—see below.)

The **Web Requests** facility can also create requestor records automatically, using the email address that a user specifies when submitting a request. The three **Automatically Create** options have the same effect on **Web Requests** as they do on MainBoss Service.

Remember that your **MainBoss Requests** license may place limits on the total number of requestors allowed in the **Requestors** table.

Automatically Create Requestors: If you checkmark **Automatically Create Requestors**, MainBoss Service attempts to create requestors based on existing records in the **Contacts** table. It checks the contacts records to see if any have an "**Email**" field matching the email address of the person who sent the request. If so, MainBoss Service creates a requestor record referring to that contact record and the email request is accepted. Otherwise, MainBoss Service checks to see if any contact record has an "**Alternate Email**" field matching the email addess of the person who sent the request. Again, if MainBoss Service finds a match, the service creates a requestor record referring to the contact record.

There may be multiple contact records whose "**Email**" or "**Alternate Email**" fields match the sender's email address. In this case, MainBoss Service can't decide between them. The email will be marked with the error `Senders email address matches multiple Contacts` and the message will be marked for rejection.

Automatically Create Requestors from Active Directory: If you checkmark **Automatically Create Requestors from Active Directory**, MainBoss Service begins by checking existing records in the **Contacts** table, going through the same process as **Automatically Create Requestors**. If a matching contact record is found, MainBoss Service will create a corresponding requestor record as described for **Automatically Create Requestors**. If no matching contact record is found, MainBoss Service then checks through any *deleted* contact records. If any deleted record has an "**Email**" or "**Alternate Email**" field matching the sender's email address, MainBoss Service will restore the deleted record (undelete it) and use it to create a requestor record. MainBoss will also restore a contact record if its "**AD Reference**" matches the requestor's Active Directory entry.

If there are no deleted contact records with the sender's email address, **Automatically Create Requestors From Active Directory** tells MainBoss Service to create a new contact record for the sender. MainBoss Service will check the Active Directory of your Windows network, and will look for any person whose primary email address matches the sender's email address. If there is a single matching address, MainBoss Service creates a new contact record using information from the Active Directory entry; if there are multiple matching addresses, MainBoss Service issues an error message (since there is no way to choose which person is correct).

If there are no matching email addresses, MainBoss Service checks the secondary email addresses in Active Directory. Again, if there is a single record with a matching email address, MainBoss Service creates a MainBoss contact record using information from the Active Directory. If there are multiple records with maching addresses, MainBoss Service issues an error message.

If MainBoss Service attempts to create a contact record, the process will fail if there's already a contact record with the same name. In this case, MainBoss Service will mark the request for rejection.

If MainBoss Service has been successful in finding or creating a contact record, the service creates a requestor record referring to that contact record. It then creates a request based on the sender's email message.

Automatically Create Requestors From Email: If you checkmark **Automatically Create Requestors From Email**, MainBoss Service begins in the same manner as **Automatically Create Requestors From Active Directory**—it first determines if there are any contact records with email addresses matching the sender's address. If not, MainBoss Service checks for deleted contact records; if any matches are found, MainBoss Service restores the record and proceeds just as with **Automatically Create Requestors From Active Directory**.

If there are no matching contact records (active or deleted), then MainBoss Service attempts to create one. In order to do so, it begins in the same way as **Automatically Create Requestors From Active Directory**: it searches the Active Directory for an entry whose primary email address matches the sender's email address. The Active Directory information is used to create a contact record if there is one and only one Active Directory entry whose primary email address matches the sender's. If MainBoss Service can't find an Active Directory entry with a matching primary email address, it checks secondary email addresses. Again, if there is one and only one matching address, MainBoss Service creates a contact record using the Active Directory information; however, the "**Email**" address of this contact record will be set to the sender's email address, not to the primary email address given in Active Directory.

If no matches are found in the Active Directory, MainBoss Service will attempt to create a contact record using information in the email message itself. The contact record's "**Code**" field will be set to the message's "Display Name" and the contact record's "**Email**" field will be set to the message's sender.

When MainBoss Service attempts to create such a contact record, the process may fail (typically because there's already a contact record with the same name). In this case, the original email message is marked for rejection. If MainBoss Service succeeds in creating the contact record, it creates a corresponding requestor record and accepts the email message as a request.

In general, we do not recommend checkmarking **Automatically Create Requestors From Email**—for example, if you receive spam in your mailbox, MainBoss Service will attempt to create a requestor record for whoever sent the spam. However, the option may be acceptable if your mailbox is well protected from spam, or if you only activate the option for a limited time, e.g. when you first start using MainBoss and you want MainBoss to "learn" about your usual requestors.

Accept Auto Create Email Pattern: Is a string specifying a pattern (also called a *regular expression*). When MainBoss Service processes an incoming email request and the requestor's email address isn't already in the list of authorized requestors, MainBoss checks the sender's email address against this pattern. If the address matches the pattern, the incoming email message will automatically be accepted; in addition, the message's sender will be added to the list of acceptable requestors (if the sender isn't on the list already).

One common type of pattern will be something like "`@ourcompany.com`". This tells MainBoss Service to accept all requests from anyone whose email address includes the string `@ourcompany.com`. In other words, you'll automatically accept requests from people inside your company.

The rules governing the pattern are the standard regular expression rules for Windows. For an introduction to these rules, see

`https://www.codeproject.com/Articles/9099/The-Minute-Regex-Tutorial`

Reject Auto Create Email Pattern: Is the opposite of "**Accept Auto Create Email Pattern**"—MainBoss Service automatically rejects all email requests sent by people whose email address matches "**Reject Auto Create Email Pattern**".

For example, suppose that you have been receiving nuisance complaints from `joe@xyz.com`. You could add this address to your reject pattern so that MainBoss Service would automatically reject messages from the given address. If you have multiple addresses that you want to reject, you can just separate them with "`|`" characters as in

```
joe@xyz.com|pat@hij.net|chris@lmn.org
```

There is no limit on the length of the string you can specify.

"**Reject Auto Create Email Pattern**" overrides the **Automatically Create** checkboxes described earlier and also overrides "**Accept Auto Create Email Pattern**". If an email sender matches the "reject" pattern, the message is immediately rejected (without being tested against the "accept" pattern) and MainBoss does not try to create a requestor record for the sender.

Comments: Any comments you wish to record.

Incoming Mail section: Controls how MainBoss Service processes requests sent in by email.

Process Requestor Incoming Email: If this box is checkmarked, MainBoss Service will operate normally. If the box is blanked out, MainBoss Service will not process incoming requests. In general, this box should be checkmarked; however, if you're changing your configuration, you may wish to turn off request processing until you've completed reconfiguration. You may also want to blank out the box if you want MainBoss to stop receiving email requests.

Mail Server Type: Choose one of the options listed. If you choose **Automatically determined**, MainBoss will test each possibility in the following order: POP3 with encryption, IMAP4 with encryption, POP3S, IMAP4S, plain POP3 (no encryption), and plain IMAP4 (no encryption). In doing these tests, MainBoss only looks at the default port for each server type; if your mail server uses a different port than the default, you must mark a specific server type (e.g. IMAP4S) and specify the port number in "**Override Default Port**".

Encryption: Choose one of the options listed:

Require Encryption with a Valid Certificate: MainBoss will only deal with your mail server if it has a valid security certificate. This automatically implies that encryption will be used. You'll receive an error message if the server has an invalid certificate.

Require Encryption: MailBoss will only deal with your mail server using one of the encrypted techniques. You'll receive an error message if none of the techniques work. Note that this option requires that your mail server has a security certificate, but it accepts self-signed certificates; it also accepts certificates that are not technically valid.

When Available: If you select this option, MainBoss still tries to use encryption when dealing with your mail server. However, if the server doesn't accept encryption, MainBoss will not issue an error message; it will simply communicate with the mail server using unencrypted data.

Mail Server: The name of the computer that receives your incoming mail. For example, if you're running Microsoft Exchange, this is usually the name of your Exchange server. This machine must allow mail access through POP3 or IMAP4.

Override Default Port: Should only be filled in if your mail server uses a non-standard port. MainBoss considers the following to be the standard ports:

- For POP3 and TLS within POP3: 110
- For POP3S: 995
- For IMAP4 and TLS within IMAP4: 143
- For IMAP4S: 993

Do not specify a port number if you have marked the **Automatically determined** option under “**Mail Server Type**”.

Override Default MailBox: The name of the mailbox, if you’re using any version of IMAP4 and if you’re using any other name than the default `INBOX`.

Although the IMAP4 standard calls this the mailbox name, most people would likely call this value a folder name. In general, the only time a value should be entered here is if incoming request emails somehow end up stored in a different folder from the default. (This can happen if you’re using server-side rules to sort incoming mail into folders.)

Since “**Override Default MailBox**” is only used with versions of IMAP4, MainBoss will not try to use POP3 or any variant of POP3 if you enter a name in the “**Override Default MailBox**” field.

Mail User Name: The name of the email account to which requests will be sent. For example, if users send mail to `workreqs@yourcompany.com`, the **Mail User Name** would usually be `workreqs`.

Important Note: Different sites will have different mail packages handling POP3 and IMAP4 processing. Some packages issue an authentication error if “**Mail User Name**” includes the `@sitename` part of an email address—you must only give the user name, e.g `workreqs`. Other packages *require* the `@sitename` part—you must specify the whole email address, e.g. `workreqs@yourcompany.com`. If you are getting authentication errors with a given user name, try changing the name to the other format to see if this corrects the problem.

Mail User’s Password: The password (if any) for the specified **Mail User Name**. As a protection measure, MainBoss encrypts this password before storing it in MainBoss’s database; therefore, when you type in the password, you’ll only see “`*`” characters, the same as with most Windows passwords.

Wakeup Interval: Dictates how often MainBoss Service will check for new incoming messages. By default, this is `0:30` (every 30 minutes).

Maximum Mail Size: The maximum size allowed for any part of an incoming mail message (specified as a number of characters). For example, this limits the size of attachments that you’ll accept. If you leave this blank, MainBoss chooses a default equal to the largest number that can be stored in a 32-bit integer. (This comes out to about 4 GB.)

Manual Processing Time Allowance: If MainBoss Service marks an email message for rejection, the service doesn’t send a rejection to the sender right away. Instead, MainBoss Service delays, allowing time for human users to deal with the problem. For example, if there’s some reason that MainBoss Service can’t create a requestor record for the sender, MainBoss waits a certain length of time; this gives you a chance to resolve the problem, e.g. by manually creating a requestor record for the sender.

The length of time that MainBoss waits is dictated by the value of "**Manual Processing Time Allowance**". For example, if you fill in this field with `12:00:00`, MainBoss will wait 12 hours after an email request is marked for rejection. This gives you time to do something about the request, if you choose to do so.

If the time allowance expires and nothing has changed, MainBoss will reject the request.

If the error message changes on a received request, MainBoss starts on a new delay. For example, suppose you see that there's a problem with a request and you make some change that you think will fix the problem. However, when MainBoss tries to process the request, a different problem arises. In this case, MainBoss will wait the full delay time again before rejecting the request; this gives you a chance to do something about the new problem.

Outgoing Mail section: Controls how MainBoss Service sends out email (acknowledgements to requestors and notifications to assignees).

Process Notification Email: If this box is checkmarked, MainBoss Service will send out notifications as normal. If the box is blanked out, MainBoss Service will not send out notifications. In general, this box should be checkmarked.

Return Email Address: The return email address to be placed on acknowledgements and notifications (i.e. the address that will be used if the recipient wants to reply). This should be the email address of a person in the maintenance department who can personally handle responses from clients.

Important: The "**Return Email Address**" should *not* be the same as the mailbox to which users submit requests. If it is, you'll run into trouble if a requestor has an auto-reply set up, such as "Hi, I've gone on vacation and won't be answering my email." What happens is this: if MainBoss Service sends an acknowledgement to the user, the auto-reply replies to the MainBoss Service mailbox; MainBoss Service thinks this is a new request, so sends an acknowledgement to the requestor; the requestor's auto-reply sends another reply to MainBoss Service; MainBoss Service sends another acknowledgement; and so on, until someone runs out of disk space. The way to avoid this is to set "**Return Email Address**" to refer to a human being (who can be smart about dealing with auto-replies).

Return Email Display Name: The name to be associated with "**Return Email Address**".

HTML Email Notification: If this is checkmarked, MainBoss Service sends notifications in HTML format. Otherwise, notifications are sent as plain text.

MainBoss Web Access URL: If you have a **Web Access** license, MainBoss Service can add HTML links to notifications that will let users jump from the notification email directly to a web page displaying an associated request or work order. In order to do this, you have to specify the URL of the web site that you use for MainBoss's **Web Access**. For example, you might set "**MainBoss Web Access URL**" to

`http://ourcompany.com/MainBossWeb`

Notification Interval: Dictates how often MainBoss Service will check to see if notifications should be sent. By default, this is `0:10` (every 10 minutes).

SMTP Server: The name of the computer that will handle MainBoss Service's outgoing mail. Normally, this is the same as "**Mail Server**" in the **Incoming Mail** section of the configuration record; however, it doesn't have to be.

SMTP Port: The port used by the SMTP mail server. This is usually port 25 or 587.

Use TLS/StartTLS: If this box is checkmarked, MainBoss Service will use TLS encryption when sending out mail.

SMTP Credential Type: The options in this area determine what kind of authentication (if any) MainBoss Service will use when sending out mail.

Anonymous: If this option is selected, MainBoss Service will not use authentication (mail is sent without going through any identification process).

Use the default network credential: If this option is selected, MainBoss Service uses whatever authentication is provided by the network as the default.

Using the specified SMTP domain, username and password: Choosing this option indicates that your SMTP server requires a domain, username, and password in order to send out mail. The fields below specify the required information:

SMTP User Domain: The domain that MainBoss Service should specify when sending out mail.

SMTP User Name: The username that MainBoss Service should specify when sending out mail.

SMTP Encrypted Password: As a protection measure, MainBoss encrypts this password before storing it in MainBoss's database.

Messages section: Controls the text strings used by MainBoss Service when sending email to users.

The main list in this section shows the strings whose text can be changed. For example, the string with identifier `EstimatedCompletionDate` is used as a label when MainBoss Service puts an estimated completion date into an email message. If you change the value of this string to `"We think the job will be done by:"`, MainBoss will use the given phrase instead of the default `"Estimated Completion Date:"`.

You can set up different strings for different languages. For example, if some of your users and/or requestors prefer to use English and others prefer to use French, you can create a set of English strings and a separate set of French strings. (Note: you specify people's preferred language in their **Contacts** record.)

Note: Before MainBoss 4.0, a different system was used to customize the contents of email messages. If you update from an old version of MainBoss to a version that uses the system described here, the updating process will automatically attempt to incorporate any existing customizations you have made.

After you have created or editing your configuration information and saved the configuration record, go to **Administration** | **MainBoss Service** and click **Manually run the MainBoss Service** (on the second drop-down button at the bottom of the window). This performs a single trial run of the MainBoss Service software; it produces a number of diagnostic messages which are displayed in the log message window immediately above the button. If there are problems with your configuration information, the log messages should help you make corrections.

You can use **Manually run the MainBoss Service** even if you don't have Windows Administration privileges. **Manually run the MainBoss Service** *doesn't* actually run a Windows service, but it runs the same software that will be used for MainBoss Service later on. Output from this software is written to the MainBoss Service log and possibly to error windows.

If the log messages that you see don't tell you enough, you may gain more information by going to the Windows event logs. However, Windows Administration privileges are required to view the event logs.

Installing MainBoss Service

In order to install MainBoss Service, you must be logged in to Windows under a login name that has both Windows Administration privileges and SQL Server Administration privileges.

Note: You should only install MainBoss Service after you have used **Manually run the MainBoss Service** to test your configuration and are satisfied that the configuration works the way you want.

Start MainBoss and go to **Administration** | **MainBoss Service**. Click the button **Configure the Windows Service for MainBoss**. This actually installs the service as a Windows service, using the configuration information specified in **Administration** | **MainBoss Service** | **Configuration**. The service will be installed to run under the `Network Service` login name on the computer where you are currently running MainBoss.

Note that you can install MainBoss Service on any computer in your network, provided that the computer has access to the instance of SQL Server that manages your MainBoss database.

The installation process accesses SQL Server in order to give MainBoss Service appropriate SQL Server permissions for accessing the MainBoss database. (This is why you must have

SQL Server Administrator permissions to install MainBoss Service.) See *SQL Server and MainBoss Service* on page 76 for information about how MainBoss Service identifies itself to SQL Server.

MainBoss Service can also be installed from a command line, rather than using MainBoss itself. For more information, see *Invoking MainBoss Service from a Command Line* on page 39.

Access to the MainBoss Database: MainBoss Service must access the MainBoss database in order to do some of its work. This means it must go through the instance of SQL Server that manages the MainBoss database, and it must have appropriate permissions to do so.

MainBoss Service inherits its SQL Server permissions from the person who installs the service.

- If you install MainBoss Service from inside MainBoss itself, MainBoss checks your record in the **Users** table. This record can specify either Windows Authentication on `Network Service`, or a SQL Server name-and-password; see *Registering New Users with MainBoss* on page 14 for more information. MainBoss Service is therefore installed to access the MainBoss database under your Windows login name or with the SQL Server name-and-password that you use.
- If you install MainBoss Service from the command line, the default is for MainBoss Service to access the MainBoss database using Windows Authentication from `Network Service`. However, you can specify different authentication information on the command line: a SQL name and password.

We recommend using Windows Authentication with `Network Service` unless there's some good reason to do otherwise. In particular, we've seen cases where MainBoss Service stopped working because it was set up to access the database using the SQL name-and-password associated with a particular IT person. When that person left the organization, the name and password were deleted and suddenly MainBoss Service couldn't access SQL Server anymore.

Deleting MainBoss Service

You may wish to delete MainBoss Service (for example, to change the computer on which it's running). In order to do the deletion, you must be logged in on the computer where the service is running, on an account with Windows Administration privileges.

To uninstall the service, start MainBoss and go to **Administration** | **MainBoss Service**. Click the second drop-down arrow at the bottom of the screen, then click **Delete Windows Service for MainBoss**.

The MainBoss Service Log

The MainBoss Service log lists any diagnostic messages that MainBoss Service has received. If MainBoss Service isn't behaving the way you expect, checking the log may help you determine what's going on. The log is displayed at **Administration** | **MainBoss Service**.

Note: If MainBoss Service encounters such serious problems that it can't start properly, it can't write diagnostic messages into the log. In this case, you may find it helpful to consult the main Windows event logs. If you are trying to start MainBoss Service on a different computer, check the event logs of both that computer and the computer you are working on. You will need Windows Administration privileges to see these logs.

The fourth drop-down button at the bottom of **Administration** | **MainBoss Service** offers a number of options that control how much information is recorded in the log. By selecting options that are relevant to your problems, you may obtain messages that help you correct what's going on. You may also choose to delete log entries, in order to clear the existing messages and make it easier to find new messages.

It may be useful to use Google or another search engine to search for explanations of the messages.

Don't confuse the MainBoss Service log with **Administration** | **MainBoss Service** | **Email Requests**. The **Email Requests** table shows incoming email messages that were turned into requests.

Normally, the log is automatically updated whenever an event occurs. In some cases, however, you may need to click the **Refresh** button.

Invoking MainBoss Service from a Command Line

You may find it convenient to invoke MainBoss Service from a command line in a Command Prompt window—it may be simpler to test and/or create the service from a command line than from inside MainBoss itself. Invoking MainBoss Service from a command line also lets you automate certain actions in batch files and command scripts; some organizations find that this is a good fit with other standard procedures. Finally, you may be running a server on a computer that doesn't have a graphic user interface, in which case a command line interface is necessary.

In order to run MainBoss Service on a computer without a graphic user interface, use XCOPY to copy the entire MainBoss program file directory to the desired computer.

To invoke MainBoss Service from a command line, open a Command Prompt window and `CD` to the Windows folder that contains the MainBoss software. (You specify this folder when you install MainBoss.) The MainBoss Service software is stored in a file named

```
Thinkage.MainBoss.Service.exe
```

To start the service running, you use a command line of the form

```
Thinkage.MainBoss.Service option option option ...
```

Options may have any of the following forms:

`/keyword:value`
: Specifies a value. For example, `/ServicePassword:password` specifies a password.

`+Option`
: Turns on an option. For example, `+TestConfiguration` tells to test the current MainBoss Service configuration.

`-Option`
: Turns off an option.

Option Abbreviations: Option keywords on the MainBoss Service command line may be abbreviated. In this guide and other MainBoss documentation, the letters that *must* be included are shown in upper case; any or all other letters may be omitted. For example, the following are all valid abbreviations for the option `/DatabaseName`

```
/databasename
/dbname
/dn
```

Although the documentation shows some letters in upper case and some in lower, you can type letters in any case you want. For example, `/DBN` is exactly the same as `/dbn` or `/Dbn` when used on the command line.

Available Options: The following options may be used on the MainBoss Service command line.

`/DatabaseServer:name`
: The database server where the MainBoss database is hosted.

`/DatabaseName:name`
: The name of the database containing the MainBoss data on the database server.

> MainBoss Service obtains configuration information from a specific MainBoss database. For most operations, you *must* specify `/DatabaseServer` and `/DatabaseName` so that MainBoss Service can locate the database and find the required information.

`/ServiceName:name`
: The name of the service. By default, this is `MainBossService`.

`/Connection:string`
: Specifies a SQL Server connection string if necessary.

`/Version:identifier`
: Used when the version of MainBoss Service that is currently running is different from the `Thinkage.MainBoss.Service.exe` that you want to use. Typically, you only need to

specify a version if you are updating an older version of MainBoss Service to match a new release of MainBoss.

`/Authentication:WindowsAuthentication`

Used when the service should use Windows Authentication to access the MainBoss database. The `/ServiceUsername` and `/ServicePassword` give the name and password for the Windows login name that MainBoss Service should use.

`/ServiceUsername:name`

The login name for the service to run under. The default is `Network Service`, so normally, this option doesn't need to be supplied.

`/ServicePassword:password`

The password for the login name under which the service runs. This is only needed if the service doesn't run under `Network Service`.

`/Authentication:SQLPassword`

Used when the service should use a SQL name-and-password to access the MainBoss database. The `/SQLUsername` and `/SQLPassword` options give the name and password that MainBoss Service should use.

`/SQLUsername:name`

A SQL Server name that MainBoss Service should use when accessing the MainBoss database. This option should only be used in conjunction with `/Authentication:SQLPassword`.

`/SQLPassword:password`

A password that MainBoss Service should use when accessing the MainBoss database. This option should only be used in conjunction with `/Authentication:SQLPassword`.

`/TestConfiguration`

Runs the MainBoss Service software to test the configuration information stored in the associated MainBoss database. This doesn't attempt to perform any actual processing; it just tests the validity of the configuration information and produces any appropriate diagnostic messages.

If you already have a version of MainBoss Service running, diagnostic messages issued by that MainBoss Service may duplicate the ones produced by running the software with `/TestConfiguration`.

`/ManuallyRun`

Runs the MainBoss Service software as a normal application program rather than as a Windows service. This is the next step beyond `/TestConfiguration`: it doesn't just test the configuration information, it actually attempts to do a single round of processing incoming requests and outgoing acknowledgements. Once it finishes this processing, the software terminates.

If you already have a version of MainBoss Service running, diagnostic messages issued by that MainBoss Service may duplicate the ones produced by running the software with `/ManuallyRun`.

`/CreateService`

Creates a Windows Service that runs MainBoss Service. You need Windows Administration privileges to make this work. The command line should specify `/DatabaseServer` and `/DatabaseName` so that the software can find configuration information for the service. The options `/ServiceUsername` and `/ServicePassword` may also be needed if you don't want to execute the service under `Network Service`. Finally, you must specify `/ServiceName` if you don't want to use the default name of `MainBossService`.

`/DELeteService`

Deletes an existing copy of MainBoss Service that is currently running as a Windows service. You need Windows Administration privileges to make this work. If you have multiple services running, use `/ServiceName:name` to specify which service you want to delete.

`/GrantAccess`

Requires SQL Server Administrator privileges on the SQL Server that manages the MainBoss database. If you have those privileges, `/GrantAccess` sets up SQL Server permissions so that the login name specified by `/ServiceUsername` can access the MainBoss database. The `/GrantAccess` command line should include the options `/ServiceUsername`, `/DatabaseServer` and `/DatabaseName`.

`/Force`

Forces an action to take place, even if errors or other conditions would normally prevent the action from happening. This is sometimes necessary if the computer where the service should be running is no longer available.

`/Update`

Updates an older version of MainBoss Service. This is equivalent to deleting the existing service and creating a new one, using the software in `Thinkage.MainBoss.Service.exe`.

`/ListMainBossServices`

Lists all the MainBoss Services currently running as services on the current computer.

Notes: Using the command line interface to MainBoss Service is the only way to install it under any other login name but `Network Service`. This interface also lets you install MainBoss Service to run on a computer where MainBoss itself is not installed. However, even if MainBoss itself is not on the computer, the software for MainBoss Service must be on the computer.

Transferring a Database to a New Computer

When you install MainBoss Service on a computer, MainBoss stores various configuration information in the database. Amongst other things, this information tells MainBoss that the service is installed.

Suppose you want to move your MainBoss database to a different computer, or you move your SQL Server to a different computer. Ideally, you should do this by following these steps:

1. Have all users on all computers exit the MainBoss program.
2. On the original server computer, login to an account with Windows Administrator privileges.
3. If you are running MainBoss Service, uninstall it as described in *Deleting MainBoss Service* on page 38.
4. Start up MainBoss and backup the database using **Administration** | **Backups** (as described in *Backups within MainBoss* on page 50).
5. Exit MainBoss.
6. Copy the backup file to the new server computer.
7. Use **Create New Organization from Backup** to create a new MainBoss database from the backup file (as described in *Restoring a Backup Within MainBoss* on page 53).
8. Start MainBoss Service on the new server computer (as described in *Setting Up MainBoss Service* on page 25).

This is the recommended approach for transferring a database to a new server or when you transfer SQL Server to a different computer. In some cases, however, you may be doing the transfer because the old server computer is unusable (e.g. due to hardware failure). In this case, you don’t have a chance to uninstall MainBoss Service first, and you’ll be working from an old backup. The old backup database contains records stating that the service is still installed, even though the service hasn’t been installed on the new computer. To correct this situation, follow these steps:

1. Make a copy of the most recent backup of your database from the old server computer.
2. Use **Create New Organization from Backup** to create a new MainBoss database from the backup file (as described in *Restoring a Backup Within MainBoss* on page 53).
3. Start MainBoss and go to **Administration** | **MainBoss Service**.
4. On the second drop-down button at the bottom of the screen, click **Delete Windows Service for MainBoss**. (It will probably take a long time for this operation to finish.)
5. Go to **Administration** | **MainBoss Service** | **Configuration** and make any changes that are needed to work on the new computer.
6. Once you have saved your new configuration, start MainBoss Service as described in *Setting Up MainBoss Service* on page 25.

If the computer that was running the old MainBoss Service ever comes back into operation, you may need to delete the old service using `sc delete`. If you do not, the old service will start up every time Windows does. However, it will then check the MainBoss database and see that the database is no longer configured to use the old service. The old service will then issue an error message and terminate itself. Therefore, the old service won’t interfere with MainBoss operations; however, it will add overhead to your system startup, so you should still get rid of it with `sc delete`.

MainBoss Service and MainBoss Upgrades

Before installing a new update of MainBoss, we strongly recommend that you use the old version of the software to stop MainBoss Service before you install the new software. Once it’s stopped, you can install the new software (including the new version of MainBoss Service). Once MainBoss is working again, you can restart the service (as described in *Installing MainBoss Service* on page 37).

Customizing and Translating MainBoss Service Messages

MainBoss uses MainBoss Service to send email messages to users and requestors. For example, when a requestor submits an email request, MainBoss Service places the request in the MainBoss database, then sends the requestor an email message saying that the request has been received.

You can customize the messages that MainBoss Service sends. For example, if you want to invite your clients to fill out a user satisfaction survey, you can add appropriate text to the standard messages that MainBoss uses.

You can also create translations of messages into the languages your clients use. In order for this to work, each person’s record in the **Contacts** table should specify his or her preferred language. Then, when an email message should be sent to that person, MainBoss will look up his or her preferred language and attempt to send the message in that language.

In order to customize messages or provide translations into other languages, you go to **Administration** | **MainBoss Service** | **Configuration** and double-click the configuration record to begin editing it. Go to the **Messages** section of the record; you’ll see a list of all the message strings you can customize. Click on one, then click 🗋 to begin the process of customizing the string or adding a new translation.

For more information, press **<F1>** to see the online help.

Using Multiple Maintenance Organizations

Some maintenance departments may decide to license multiple maintenance organization databases. For example, a property management company that services multiple independent properties may choose to have a separate database for each property.

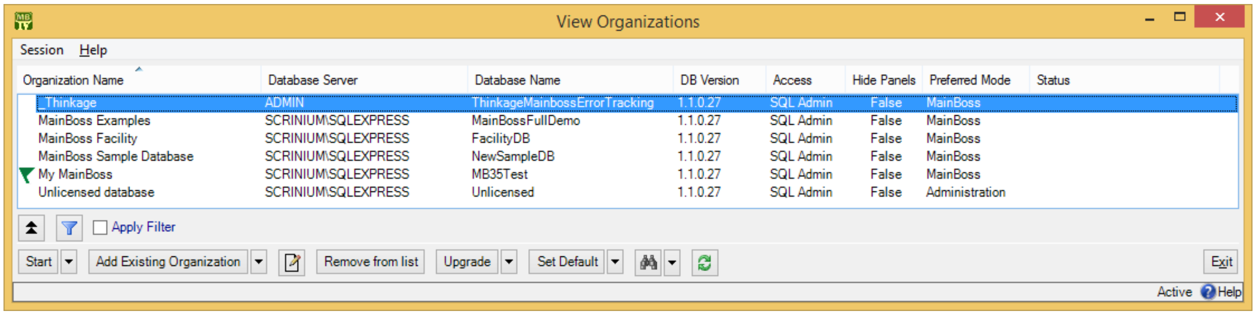
If you have multiple databases, each must have its own separate set of license keys. You may not use the same license keys for multiple databases.

1. You create each new database following the same procedures described in *Creating a Maintenance Organization* on page 7.
2. You then record the database's license keys as described in *Entering License Keys* on page 9.
3. You authorize appropriate users (as described in *Users* on page 13).
4. On every computer where someone will wish to access a particular database, you follow the steps in *Installing MainBoss on Other Computers* on page 22 so that the databases are known on each appropriate computer.
5. If you are using MainBoss Service, each separate database must have its own email address for receiving mailed-in requests. You must also install a separate MainBoss Service for each database where MainBoss Service will be used. If multiple MainBoss Services run on the same computer, each one must have a different "**Service Name**". For more information, see *Setting Up MainBoss Service* on page 25.

You will end up with a list of maintenance organizations you may access. One organization is designated as your personal default—whenever you start MainBoss on a particular computer, that computer will start with your default organization on that computer. To switch to another maintenance organization, go to MainBoss's **Session** menu and select **Change Maintenance Organization**. MainBoss will display a list of available organizations.

□ To switch to another maintenance organization:

1. Go to MainBoss's **Session** menu and select **Change Maintenance Organization**. MainBoss will display a list of available organizations:



Organization Name	Database Server	Database Name	DB Version	Access	Hide Panels	Preferred Mode	Status
_Thinkage	ADMIN	ThinkageMainbossErrorTracking	1.1.0.27	SQL Admin	False	MainBoss	
MainBoss Examples	SCRINIUM\SQLEXPRESS	MainBossFullDemo	1.1.0.27	SQL Admin	False	MainBoss	
MainBoss Facility	SCRINIUM\SQLEXPRESS	FacilityDB	1.1.0.27	SQL Admin	False	MainBoss	
MainBoss Sample Database	SCRINIUM\SQLEXPRESS	NewSampleDB	1.1.0.27	SQL Admin	False	MainBoss	
My MainBoss	SCRINIUM\SQLEXPRESS	MB35Test	1.1.0.27	SQL Admin	False	MainBoss	
Unlicensed database	SCRINIUM\SQLEXPRESS	Unlicensed	1.1.0.27	SQL Admin	False	Administration	

2. Select the organization you want, then click **Start**. MainBoss will open the organization and you can begin working with it.

If you wish to change the default, select the desired database in the above window and click **Set Default**. This designates the chosen database as the default. From this point on, MainBoss on this computer will start with the selected default. (Note that each user on each computer can have a different default.)

Organization lists are specific to a particular person on a particular computer. You may have different organization lists on different computers. Also, each person on a particular computer may have his or her own personal organization list.

Start Modes

MainBoss can start in a number of different modes. To start in a specific mode, go to your organizations list and use the drop-down arrow associated with the **Start** button. The following modes are available:

MainBoss: Full MainBoss mode with all available modules operational. Your license limits the maximum number of names that may be entered in the **Users** table.

MainBoss Assignments: Only displays the control panel entry for **My Assignment Overview**. This lets you see in-progress requests, open work orders, and issued purchase orders that have been assigned to you. Depending on your security roles, you may also be able to see unassigned requests, work orders, and purchased orders.

MainBoss Requests: Requests-only mode. The control panel will only display entries relevant to dealing with requests. You can always open MainBoss in this mode, provided you have a **MainBoss Requests** license key and you have been granted permissions to access the database with a Named user license.

Administration: Only allows access to the **Administration** part of the control panel. You can always start MainBoss in this mode, even if you currently have no valid license keys. In fact, when you create a blank organization, you *must* start MainBoss in this mode in order to enter your license keys. Once you've done so, you can quit and start MainBoss again in its normal mode.

Administration functions can only be used if you have the `Administrator` security role.

View Sessions: Only shows who else is using MainBoss.

You can specify a start mode in the **Select MainBoss Organization** window. You can set a default mode for any organization by clicking the edit button and editing the database's entry or by choosing a mode from the **Start** drop-down button.

As an example of the use of start-up modes, your help-desk personnel might set their starting mode so that they always access the database in requests-only mode unless they manually open it in some other mode.

If you sometimes want to open a database in full mode and sometimes in requests-only mode, you can give the same database multiple organization names. For example, suppose your database has the "**DataBaseName**" `MyMainBoss`. You could use **Add Existing Organization** to create two different lines in the organization list:

- One line might have the "**Organization Name**" `MyMainBoss-Full`, with a start mode of **MainBoss**.
- A second line might have the "**Organization Name**" `MyMainBoss-Requests`, with a start mode of **MainBoss Requests**.

If you click **Start** on `MyMainBoss-Full`, MainBoss opens in full **MainBoss** mode. If you click **Start** on `MyMainBoss-Requests`, MainBoss opens in requests-only mode.

You can also use command line options to specify start mode and maintenance organization you wish to access. This makes it easy to create shortcuts and desktop icons that each open a different organization or open the same organization in different modes. For more on command line options, see the next section.

Command Line Options

If you frequently access different maintenance organization, you might want to create a set of desktop icons, each of which starts up MainBoss with a different organization. To create such icons, you need to use MainBoss command line options. These options are described in full detail in the online help, but this guide offers a quick overview.

When MainBoss starts, it usually opens your default database. To tell it to open a different database, use the command line:

`mainboss /on:`*organizationname*

For example,

```
mainboss /on:OurOrganization
```

opens the database that has the organization name `OurOrganization`.

Start Modes on the MainBoss Command Line

The command line lets you specify a particular *mode* in which you want MainBoss to operate. For example, you can start MainBoss in "requests only" mode—this lets the user view and enter requests, but nothing else. Help-desk personnel might run MainBoss in this mode so they can record problem reports but not use other MainBoss features.

To start in requests mode, use the command line

```
            mainboss /mode:requests
```

You can also add options specifying an organization name, as shown previously.

Culture Information on the MainBoss Command Line

The `/MessageCultureInfo` and `/FormatCultureInfo` options lets you specify Windows display language“culture” and formatting “culture” to be used when you execute MainBoss. A *culture* is a collection of information about language and other locale-based information, e.g. what monetary symbols are used in a particular place.

By default, MainBoss takes its message culture information from the current Windows Display Language for the user, and the formatting culture information from the “**Format**” field in the **Format** section of your Windows Regional Settings. It then attempts to operate in the language given by those settings (e.g. MainBoss attempts to run in French if that’s what your settings dictate). However, if you specify `/MessageCultureInfo` and `/FormatCultureInfo` arguments on the command line, the argument overrides the Regional Settings. This is particularly useful if you’re creating a batch command file or shell script that you want to have work in a specific language, no matter what the user’s Regional Settings might be.

The argument of `/MessageCultureInfo` and `/FormatCultureInfo` is the official Windows “CultureInfo” string name for the desired culture. For example, if you want generic Spanish, you’d specify `/MessageCultureInfo:es-ES`; for U.S. English, you’d use `/MessageCultureInfo:en-US`. See your Windows documentation for recognized “CultureInfo” string identifiers.

The most common formatting culture “override” users may require is the currency symbol. This is configured in the culture specified in the “**Format**” field in the **Format** section of your Windows Regional Settings. If you override the currency symbol, then this symbol will be used in MainBoss reports and screen displays for currency.

Options on the MainBoss command line are not affected by `/MessageCultureInfo`. For example, the option `/Mode:Requests` does *not* change if you switch to a language where “requests” is translated to a different word.

Note: If you change the message culture information with `/MessageCultureInfo`, MainBoss makes the change *after* the command line arguments have been parsed. This means that if your command line has syntax errors, you may see error messages in the user’s original preferred language, not in the language specified by `/MessageCultureInfo`.

Invoking Compact Browsers with the MainBoss Command Line

The `+CompactBrowsers` option invokes MainBoss in a mode that initially hides the details panel for records selected in table viewers. The `-CompactBrowsers` option tells MainBoss not to hide such panels.

Creating Desktop Icons that Use Command Line Options

The easiest way to use command line options is to create a desktop icon that specifies the options. You can do this by following these steps:

1. Right-click an open area of the Windows desktop. In the resulting menu, click **New**, then **Shortcut**.
2. Windows asks you to specify the file to execute. Click the **Browse** button and use it to locate the MainBoss executable file (named `mainboss.exe` in the folder where you installed the MainBoss software—typically `\Program Files\Thinkage\MainBoss\4.0`).
3. Click **Next**.
4. Enter a descriptive name for the shortcut, then click **Finish**. This creates a new desktop icon with the name you specified.
5. Right-click on the new desktop icon. In the resulting menu, click **Properties**. Windows will open a Properties window for the icon.
6. The "**Target**" field contains the name of the MainBoss executable file in double quotes. After the closing double quote on the file name, add a space and whatever options you wish to specify.
7. Click **OK**.

The icon that you just created will start up MainBoss with the command line options you specified. You can have multiple MainBoss desktop icons, each with a different command line. (Each such icon has to be given a unique name.)

Backups

We strongly recommend that you backup your database on a regular basis. Most commercial backup software packages can tell SQL Server to create a backup of a database, after which the backup package makes the backup. In this way, backing up your MainBoss database is automatically incorporated into your normal backup procedures.

> If you use commercial backup software, check that it can, in fact, deal with SQL Server databases. With some backup software products, dealing with SQL Server is an extra that you have to pay for on top of standard backup operations.

If you do not have commercial software that deals with SQL Server, you have a number of options:

- You can do backups manually through SQL Server Management Studio. For more information, see *Backups with SQL Server* on page 71.
- You can use the MBUtility command as described in *Backing Up Your Database with MBUtility* on page 66.
- You can back up your database inside MainBoss by using **Administration** | **Backups**. For details, see the next section.

Ideally, however, you should use automated backup facilities as discussed above, either through commercial software or through SQL Server's built-in facilities. Because these facilities operate automatically, you won't have to remember to do backups manually.

> We strongly recommend that you save copies of your backup files in off-site locations. For example, you might copy your backup files to a CD or DVD, then take the disk home with you; alternatively, you could copy your backup files to a computer at a company branch office or to a location in the Internet "cloud". If your only backups are in the same building as the computers that hold your original data, a fire or flood in that building could wipe out all your records.

Backups within MainBoss

You can perform backups within MainBoss using **Administration** | **Backups**. The general procedure is:

1. Create one or more *backup file names*. A backup file name is the name of a file into which MainBoss should write backup data. For example, you might create seven backup names: one for each day of the week.
2. When you want to perform a backup, select a name from the list of available backup names. Then click **Backup**. This submits a request to the SQL Server asking the server to create a backup. (In order to use this function, you must have SQL Server Administrator privileges.) When the backup occurs, it will overwrite any current contents of the specified backup file.
3. The next time you want to do a backup, select a different name from the list and click **Backup** again. MainBoss sorts the backup names so that the oldest backup file is at the top of the list. Therefore, if you have seven backup names, and if you always select the top name whenever you make a backup, your backup will always overwrite the oldest file. This makes it easy to reuse the same set of files by cycling through them.
4. When you initiate a backup operation, MainBoss makes an appropriate note in the database history (**Administration** | **Database Management**. If the backup fails for some reason, MainBoss makes note of the failure in the database history; however, if the backup

succeeds, MainBoss doesn’t make a history note. In other words, if there’s no note about the backup failing, you can assume that it succeeded.

A number of important considerations apply to the backup name:

- Backup operations are run by SQL Server itself on the computer where SQL Server is running. Because SQL Server does the backup, backup names must refer to files that are accessible to the login name under which SQL Server runs. Furthermore, backup names are interpreted relative to the computer where SQL Server is running.

  For example, suppose SQL Server is running on Computer X, but you’re working on Computer Y. When you specify the backup name,

  ```
          C:\MyBackups\Monday.bak
  ```

  you may think that this refers to a file on Computer Y’s `C:` drive. However, the backup is done by SQL Server on Computer X. Therefore, SQL Server will try to save to Computer X’s `C:` drive, not Computer Y’s. If the directory `C:\MyBackups` doesn’t exist on Computer X (or if SQL Server’s login name doesn’t have permissions to write to that folder), you’ll get an error message and the backup won’t work.

- If you specify a backup name that does not begin with “`\\`” and does not begin with a drive letter (e.g. “`C:\`”), SQL Server automatically assumes the name refers to a file under SQL Server’s working folder. The location of this working folder depends on information specified when SQL Server was installed. It is likely under `Program Files` on the computer where SQL Server runs; since the `Program Files` folder is often read-only, you may have to take special action to ensure that the backup process has write permission on the folder where backup files should be written.

- SQL Server backup files typically have names ending in the extension “`.bak`”, as in `MyBackup.bak`. Therefore, if you specify a simple name for a backup name—a name that doesn’t have any slashes, backslashes or dots—MainBoss automatically adds “`.bak`” on the end of the name. For example, if you specify a simple name like “`abc`”, MainBoss automatically turns this into “`abc.bak`”.

- For safety’s sake, backup names should refer to a disk drive that is *different* from the one that contains the actual MainBoss database. Otherwise, if the disk drive malfunctions, you could lose both your original database and your backups. It’s even better if the backup name refers to a completely different computer, as in `\\NotTheSameComputer\MyBackups\Backup1.bak`.

  It’s also a good idea to make copies of backup files and to keep those copies somewhere off your premises. That way, if you have a fire or some other problem that affects your entire site, you’ll have backups someplace else safe.

  (By default, SQL Server puts backup files on the same disk as the corresponding database. Therefore, you should make sure *not* to use the default location.)

When you specify a backup name, the “**Defaults**” line in **Administration** | **Backups** tells you the computer, the directory, and the file name extension. **We strongly recommend that you**

examine the "Defaults" line whenever you create a backup name and before you click **Backup**. This will avoid failed backups or accidentally creating a backup file somewhere you don't expect.

The information in "**Defaults**" is generated by MainBoss at the time the window is displayed. It's dependent on your SQL Server configuration. Therefore, suppose you specify a backup name of "`abc`". "**Defaults**" shows where the backup file would be written if you clicked **Backup** right now. However, if your SQL Server configuration changes, future backups with the "`abc`" backup name might be written to a different folder.

The window for viewing backup names includes the following:

Name list: As noted above, the default is to put the oldest backup file at the top of the list.

Information area: The area below the list shows information about the selected backup name/file.

Last Backup Date: The last date/time that backup information was stored in the selected file.

Database Version: Every version of MainBoss has an associated database version number. This number reflects the format of the database. It only changes when you install a new version of the MainBoss software; in some cases, two versions of the software may use the same version of the database format. In general, the version number of the database has no connection with the version number of the software.

File Name: The name you specified for the file.

Defaults: Information telling where the backup information will be written if you click **Backup** right now.

Comments: Any comments you've associated with this backup name.

Message: Any messages generated during the backup process.

New Backup Name: Opens a window where you can specify a new backup name.

Backup: Begins a backup on your database. See the notes at the start of this section for further information on the backup process.

Note: If you perform backups and restores entirely through SQL Server, MainBoss won't store information about such operations. You won't see corresponding entries in the **Backups** list or in the database history.

When MainBoss initiates a backup, it makes an appropriate entry in the database history (**Administration** | **Database Management**). If the backup fails, MainBoss makes a second entry in the database history reporting the failure. If the backup succeeds, MainBoss doesn't record this—the default is to assume that the backup was successful.

After a backup, MainBoss also records any generated messages in the **Message** section of **Administration** | **Backups**. If a backup fails, check this section for useful information.

Restoring a Backup Within MainBoss

SQL Server offers various methods for restoring a database from a backup file. For further information, see the SQL Server documentation.

However, you can also restore a database from within MainBoss itself, using facilities in the Maintenance Organization window. (This is the window that lets you choose a database to open. One way to open this window is to go to MainBoss's **Session** menu and click **Change Maintenance Organization**.)

MainBoss offers two operations for restoring databases from backup:

- **Restore Organization** (on the **Upgrade** button's drop-down list) overwrites an existing database. First you select the database to be overwritten (by clicking on the entry for that database in your Maintenance Organization list), then you click on **Restore Organization** to specify the backup file you want to use.

  Typically, you would use **Restore Organization** if you made some sort of large-scale error that you wanted to undo (e.g. mistakenly generating and committing a large number of planned maintenance work orders). **Restore Organization** is not intended for situations where your database is no longer usable because it's been deleted or corrupted.

  Restore Organization is particularly useful when you're just getting started with MainBoss. For example, you can backup your database and then experiment with various ways of setting up your work (especially planned maintenance tasks). If you decide to take a different approach, you can restore from your backup and try again.

  Restore Organization also ensures that the permissions on the database file are the ones that MainBoss needs and expects. This can be helpful if the permissions somehow get changed.

- **Create New Organization from a Backup** (on the **Add Existing Organization** button's drop-down list) creates a separate new database using data from a backup file.

  Typically, you would use **Create New Organization from a Backup** if your original database has been deleted or corrupted. You could also use this operation if you wanted to create a test database using data obtained from your production database.

In both cases, MainBoss performs the operation by submitting requests to SQL Server. You must therefore have SQL Server Administrator privileges to perform either operation.

Restore operations are run by SQL Server itself on the computer where SQL Server is running. Because SQL Server does the work, the backup file you want to use must be accessible to the login name under which SQL Server runs. Furthermore, the name of the backup file should be relative to the computer where SQL Server is running. Unless the file name explicitly begins with `\\computer` or a drive name (e.g. `C:\`), the file name will be assumed to be relative to SQL Server's home directory. (This is true even if the file name begins with a `\`.)

For example, suppose SQL Server is running on Computer X and you submit a restore request while working on Computer Y. MainBoss will ask you to specify the name of the backup file containing the database you want to restore. Suppose you give the name

```
C:\MyBackups\Monday.bak
```

You may think this refers to a file on Computer Y's `C:` drive. However, the restore operation is performed by SQL Server on Computer X; therefore, SQL Server will try to find the file on Computer X's `C:` drive, not Computer Y. If the directory `C:\MyBackups` doesn't exist on Computer X (or if SQL Server's login name doesn't have permissions to read that folder), you'll get an error message and the restore operation won't work.

For this reason, we recommend that you only submit restore requests while logged in to the computer where SQL Server is running. You should also be aware of what permissions SQL Server has and what files will actually be accessible to SQL Server's login name.

Backup Format: There are several different ways to create a backup with Microsoft SQL Server. MainBoss assumes that you did the backup using the backup facilities inside MainBoss itself, or in some other way that uses the same backup format. However, it is possible to create database backup files in such a way that MainBoss cannot restore from the file. This happens when you do the backup with SQL Server Management Studio and specify options that create backup files whose format is different than the one MainBoss expects.

In this case, you can still restore the database from the backup file. You just have to use Management Studio directly, rather than using MainBoss.

Command Format: When you click **Restore Organization**, MainBoss lists the available backup file and asks you to pick the one you want to restore. When you click **Create New Organization from a Backup**, MainBosss ask you to specify the following:

Database Server: The network location of the SQL Server through which you will access the database. If you click the accompanying "..." button, MainBoss displays a list of database servers that are known to your Windows system. There may be additional servers that you can use but that Windows doesn't currently know about.

Database Name: The name of the database you want to create. This cannot match any existing database on the given server.

Organization Name: A name that you personally will use for this database. This should not be the same as the organization name for any other database.

Backup file: Fill in the field with one of the names in your list of backup names (as given in **Administration** | **Backups**). If you choose a name from this list, MainBoss tries to restore from the associated backup file.

MainBoss's list of backup files only contains files made inside MainBoss using **Administration** | **Backups**. The list doesn't backups made directly with SQL Server Management Studio.

Backup Set number: Used only if the backup file was created using SQL Server options related to backup sets. For backups made with **Administration** | **Backups** inside MainBoss, leave "**Backup Set number**" blank.

Once you have specified the required data, click **OK**. This submits an appropriate request to SQL Server, which performs the restoration.

The MBUtility Command

The MBUtility command offers a command-line interface to selected MainBoss operations. This allows administrators and IT personnel to create shell scripts (e.g. `.bat` or `.cmd` command files) which perform common administrative functions.

MBUtility is installed as an `.exe` file in the same folder as other MainBoss software. To execute MBUtility, you can `cd` to this folder or you can create a shortcut to the program. You can also add the MainBoss software folder to your `PATH` environment variable.

> MBUtility is only available if you have installed the MainBoss package manually. The software will not be available on your computer if you use ClickOnce deployment.

In general, the format of the command is

```
mbutility verb option option option ...
```

The verb must always come immediately after `mbutility`, but the options may appear in any order.

Options may have any of the following forms:

`/keyword:value`
Specifies a value. For example, `/Output:filename` specifies an output file.

`+Option`
Turns on an option. For example, `+Probe` may be used with the `AddOrganization` verb; it tells MBUtility to check that the corresponding database is actually accessible from the current computer.

`-Option`
Turns off an option. For example, `-Probe` may be used with the `AddOrganization` verb; it tells MBUtility not to check the database's accessibility, but just to add it to the list of known organizations.

Abbreviations on the MBUtility Command Line: Most keywords on the MBUtility command line may be abbreviated. In the sections that follow, letters that *must* be included are shown in upper case; any or all other letters may be omitted. For example, the following are all valid abbreviations for the option `/DatabaseName`

```
/databasename
/databasen
/dbname
```

```
/dn
```

Although the documentation shows some letters in upper case and some in lower, you can type letters in any case you want. For example, `/DBN` is exactly the same as `/dbn` or `/Dbn` on the MainBoss command line.

Getting Help for MBUtility

The command

```
mbutility Help
```

displays a summary of MBUtility's syntax and options. As noted in the previous section, `Help` may be abbreviated by omitting any or all lower case letters, as in

```
mbutility h
```

Listing MainBoss Organizations

The `ListOrganizations` verb of MBUtility lists MainBoss maintenance organizations known to the current user (i.e. the organizations that would be listed in the **Select Maintenance Organization** window). The command has the format

```
mbutility ListOrganizations option ...
```

where the possible options are:

`+Probe`
: Determines whether you can actually connect with each organization listed. The default behavior is `-Probe`, which means that MBUtility doesn't test connections in this way.

`+RealNames`
: Displays the real registry name for each database, as well as the databases' usual display names. The default behavior is `-RealNames`, which means MBUtility only shows the usual display names.

`+OrderbyServer`
: Sorts the list by server name.

`+OrderbyDatabaseName`
: Sorts the list by database name.

`+DatabaseVersion`
: Shows the version of each database, provided that the database is accessible and its version can be read.

`+ShowDefaults`
: Shows which database is the default one to start, and which database was the last one you used (if it's difference from the default database).

`+AllUsers`

Displays the organizations in a special registry list called `AllUsers`. This list is in the HKLM section of the registry rather than a user's individual HKCU. The `AllUsers` list is particularly useful when you're using ClickOnce, since the organizations in the list will be available to all users. (However, security roles still control what a particular user can or can't do with the organization's database.) You can add organizations to `AllUsers` using the `AddOrganization` verb, and delete them from the list using `DeleteOrganization`.

Built-In Data Schemas

A *data schema* consists of XML code describing the format of data in a database. Schemas are used when importing data into a database or exporting data from the database. MBUtility has a number of built-in schemas that can be used for this purpose. To get a list of these built-in schemas, execute the command

```
mbutility ListImportSchemas
```

The result is a list of keywords that may be used on an MBUtility command line, as described in later sections of this guide.

Generating Schema Definitions: You can obtain the actual XML code for a built-in schema using MBUtility's `GenerateImportSchema` verb. The command line has the format:

```
mbutility GenerateImportSchema option option ...
```

where the options are:

`/SchemaIdentification:schema`

Specifies the schema whose definition you wish to obtain. This should be one of MBUtility's built-in schemas. Names are case-sensitive. If the schema name contains blank characters, the name should be enclosed in quotes, as in

```
/si:"Storeroom Assignment"
```

`/Output:filename`

A file where MBUtility should write the XML definition of the schema. This file will be written in Unicode characters, which means it should only be edited with software that can deal with Unicode (e.g. Wordpad).

`+EXCEL`

Specifies that the XML code in the schema should be in a format acceptable to Microsoft Excel.

For example, the following creates a schema for unit records, in a format acceptable to Excel:

```
mbutility gis /si:Unit /out:unitdef.xml +excel
```

Exporting Data from Your Database

MBUtility has several ways to export selected data from your database to an output file. For example, you might regularly wish to export data to another application for some reason; by making a command file that uses MBUtility, you could automate the export process, thereby simplifying the task (and also avoiding errors).

The command line for exporting all the tables in your database is

```
            mbutility ExportAll option option ...
```

where the options are:

`+EmbeddedSchema`
Tells MBUtility to embed data schemas for the tables in the output file. This is useful if you're exporting the data to software that accepts embedded schemas.

`-EmbeddedSchema`
Tells MBUtility not to embed data schemas in the output file. This is the default.

`+ExcludeDeleted`
Tells MBUtility not to export information from deleted records.

`-ExcludeDeleted`
Tells MBUtility to export information from deleted records.

`/Output:filename`
Specifies a file where the output should be written. This file will be written in Unicode characters, which means it should only be edited with software that can deal with Unicode (e.g. Wordpad).

`/OrganizationName:name`
The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
The name of the server that holds the MainBoss database.

`/DataBaseName:name`
The name of the MainBoss database.

If you specify `/OrganizationName`, you don't have to specify either `/DataBaseServer` or `/DataBaseName`. If you don't specify `/OrganizationName`, you must specify both `/DataBaseServer` and `/DataBaseName`.

For example,

```
            mbutility exall /out:mytables /on:MyMainBoss +es
```

exports data about units to a file named `mytables`. The output file will include embedded data schemas.

Exporting a Specific Table: The `Export` verb is similar to `ExportAll`, but it exports a single data table rather than all of them. `Export` accepts all of the options accepted by `ExportAll`. In addition, it takes the following:

`/SchemaIdentification:schema`

Specifies a data schema for exporting the data; this option is mandatory. The named schema should be one of MBUtility's built-in schemas (as discussed in *Built-In Data Schemas* on page 57). If the schema name contains blank characters, the name should be enclosed in quotes, as in

```
/si:"Storeroom Assignment"
```

`+EXCEL`

Tells MBUtility you plan to read the exported data into Microsoft Excel. In this case, MBUtility writes out *two* files: one containing the schema and one containing data that refers to the schema. Both files are written to the same directory and have the same base name (specified by the `/Output` option); however, the schema file has the file name extension `.xsd` while the data file has `.xml`. For example, if you specify `/Output:mydata`, you'd end up with `mydata.xsd` and `mydata.xml`.

`+EXCEL` cannot be used in conjunction with `+EmbeddedSchema`.

For more on using Excel with exported data, see *An Example of Exporting to Excel 2007* on page 59, *An Example of Exporting to Excel 2010* on page 60, or *An Example of Exporting to Excel 2013* on page 61.

An Example of Exporting to Excel 2007

The previous sections of this manual gave basic details of exporting MainBoss data. This section provides a concrete example, showing how to export unit data to Microsoft Excel 2007.

The first step is to export the data:

```
mbutility ex /si:unit /out:mydata /on:mymb +excel
```

This exports data from a database with the organization name `mymb`. It creates two output files: the schema is saved in `mydata.xsd` and the data itself is saved in `mydata.xml`. Both files are written to the same directory.

Now you're ready to import the data into Excel. If you have Microsoft Office 2007, here's what you do:

1. Start Excel.
2. Click the "Office" button (in the upper left hand corner of the Excel window).
3. In the resulting window, click "Excel Options" (at the bottom of the window).

4. In the resulting window, click “Popular” in the left part of the window, then checkmark “Show Developer Tab in the Ribbon” (if it isn’t already checkmarked).
5. Click OK to close the options window.
6. In the main Excel window, click “Developer” in the ribbon.
7. In the XML section of the ribbon, click “Import”.
8. In the resulting window, specify the name of the data file (in our example `mydata.xml`).

Excel gives you the option of reading the data file into the current spreadsheet or into a new one.

An Example of Exporting to Excel 2010

The previous section showed how to export to Microsoft Excel 2007. This section does the same with Excel 2010 (and repeats some of the information so that it’s all in one section).

The first step is to export the data:

```
mbutility ex /si:unit /out:mydata /on:mymb +excel
```

This exports data from a database with the organization name `mymb`. It creates two output files: the schema is saved in `mydata.xsd` and the data itself is saved in `mydata.xml`. Both files are written to the same directory.

Now you’re ready to import the data into Excel. If you have Microsoft Office 2010, here’s what you do:

1. Start Excel.
2. Click “File” in the upper left hand corner of the Excel window.
3. In the resulting window, click “Options” (at the bottom of the window).
4. In the resulting window, click “Customize Ribbon”.
5. Under “Main Tabs”, checkmark “Developer” if it isn’t already checkmarked.
6. Click OK to close the options window.
7. In the main Excel window, click “Developer” in the ribbon.
8. In the XML section of the ribbon, click “Import”.
9. In the resulting window, specify the name of the data file (in our example `mydata.xml`).

Excel gives you the option of reading the data file into the current spreadsheet or into a new one.

An Example of Exporting to Excel 2013

The previous section showed how to export to Microsoft Excel 2010. This section does the same with 2013 (and repeats some of the information so that it's all in one section).

The first step is to export the data:

```
mbutility ex /si:unit /out:mydata /on:mymb +excel
```

This exports data from a database with the organization name `mymb`. It creates two output files: the schema is saved in `mydata.xsd` and the data itself is saved in `mydata.xml`. Both files are written to the same directory.

Now you're ready to import the data into Excel. If you have Microsoft Office 2013, here's what you do:

1. Start Excel.
2. Click "File" in the upper left hand corner of the Excel window.
3. In the resulting window, click "Options" (at the bottom of the window).
4. In the resulting window, click "Customize Ribbon".
5. Under "Main Tabs", checkmark "Developer" if it isn't already checkmarked.
6. Click OK to close the options window.
7. In the main Excel window, click "Developer" in the ribbon.
8. In the XML section of the ribbon, click "Import".
9. In the resulting window, specify the name of the data file (in our example `mydata.xml`).

Excel gives you the option of reading the data file into the current spreadsheet or into a new one.

Importing Data into Your Database

The `Import` verb of MBUtility imports data into your database from an input file. For example, if you are migrating to MainBoss from some other maintenance management software package, you might be able to import some subset of the data from that other package.

Important: Importing is a complicated procedure, and should only be done by people with strong IT expertise. We strongly recommend that you backup your database before you attempt an import operation. If an error occurs during an import operation, restore the database from backup, correct the error in the import data, then try again.

The command line for importing a single table is

```
mbutility import option option ...
```

where the options are:

`/SchemaIdentification:schema`
Specifies a data schema for exporting the data. This should be one of MBUtility's built-in schemas (as discussed in *Built-In Data Schemas* on page 57). Names are case-sensitive. If the schema name contains blank characters, the name should be enclosed in quotes, as in

```
/si:"Storeroom Assignment"
```

`/Input:filename`
Specifies a file containing the data to input. This file should use Unicode characters.

`/ErrorOutput:file`
Specifies a file where MainBoss should write any error messages that arise during the import process.

`/OrganizationName:name`
The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
The name of the server that holds the MainBoss database.

`/DataBaseName:name`
The name of the MainBoss database.

If you specify `/OrganizationName`, you don't have to specify either `/DataBaseServer` or `/DataBaseName`. If you don't specify `/OrganizationName`, you must specify both of `/DataBaseServer` and `/DataBaseName`.

For example,

```
mbutility im /si:Unit /in:myfile /eo:errs /on:MainBoss
```

imports unit data from a file named `myfile`.

Important: This version of MainBoss will not let you import data that conflicts with existing records in the database. For example, if you are importing unit data, each new unit's identifier code must be different from any units that are already in the database. This means that you can't export MainBoss data, edit it with some other piece of software, then import it again to overwrite MainBoss's existing data.

ImportAll: The `ImportAll` verb imports multiple tables from a file. This file should have been created with `ExportAll` (as described in *Exporting Data from Your Database* on page 58) or with some process that creates data in the same format as `ExportAll`. `ImportAll` takes the same options as `Import`, except that you don't specify a `/SchemaIdentification`.

If you wish to create a database for testing, we recommend using backup and restore (using **Create New Organization from Backup**) rather than exporting and importing data. For more information, see *Backups within MainBoss* on page 50 and *Restoring a Backup Within MainBoss* on page 53.

Tip: Sometimes it's difficult to figure out what format to use when you are preparing data for import. When in doubt, start up MainBoss and create a record of the type you intend to import; for example, if you intend to import Unit records, create a Unit record inside MainBoss. Then export the record as described in *Exporting Data from Your Database* on page 58. The format of the exported record will be the same that's expected for imported records.

CreateMainBossBasicDatabase: The `CreateMainBossBasicDatabase` verb creates a SQL Server database containing data from a MainBoss Basic database. **This command does not create a MainBoss database.** Instead, it takes XML data that was exported from MainBoss Basic and uses that data to create a generic SQL Server database. This may be useful if you are a MainBoss Basic user and you wish to use standard SQL Server facilities to make queries on MainBoss Basic data.

The `CreateMainBossBasicDatabase` verb takes the following options:

`/DataBaseServer:servername`
The name of the server where you wish to store the created database.

`/DataBaseName:name`
The name you want to give to the new database.

`/Input:filename`
Specifies a file containing the data to input. This file should be the XML file created by exporting data from MainBoss Basic.

`/ErrorOutput:file`
Specifies a file where `mbutility` should write any error messages that arise during the import process.

Adding Organizations

The `AddOrganization` verb of MBUtility lets you add an organization to a user's organization list. By default, this will be the current user (i.e. the person who executes the `mbutility` command). However, some options let you specify a different user's organization list.

The database associated with the organization must already exist and the user must have appropriate permissions on the database. This means that the user's name must be specified in the database's **Users** table.

For example, if you are logged in under the name `jsmith`, you can use `AddOrganization` to put a new organization in `jsmith`'s list of known organizations, provided that `jsmith` appears in the database's **Users** table.

The format of the command is

```
mbutility AddOrganization option option ...
```

where the options are:

`+Replace`

This option is relevant if the user's organization list already contains an organization with the same name. `+Replace` says the old organization should be replaced with the new organization. If you do not specify `+Replace`, MBUtility will not replace old organization's with new ones; if there is a name conflict, you will be receive an error message.

`+Probe`

If you specify this option, MBUtility will attempt to verify that the user can actually access the database. If you do not specify `+Probe`, the database will be added to the user's organization list, but MBUtility doesn't check to see if the user can actually connect to the database.

`+SetDefault`

Marks the added database as your *default* database. This is the database that will automatically be opened whenever you start MainBoss.

`/Mode:startmode`

Specifies the default start-up mode when the user connects to the database. The `startmode` value must be one of the following strings:

```
MainBoss Requests
MainBoss
View Sessions
Administration
```

For explanations of these startup modes, see *Start Modes* on page 46 of this guide.

`+CompactBrowsers`

Tells MainBoss that the database should be opened in a mode that initially hides the details panel for records selected in table viewers.

`/AuthenticationType:type`

Specifies the type of authentication to be used with SQL Server when `mbutility` accesses the database. The `type` value must be one of the following strings:

```
WindowsAuthentication
SQLPassword
ActiveDirectoryPassword
ActiveDirectoryIntegrated
```

By default, `mbutility` uses Windows authentication, using your user name (i.e. the name of the person who invokes the `mbutility` command).

`/AuthenticationUsername:name`

Specifies a username that `mbutility` should use when accessing the database through SQL Server. This option must be specified if you wish to access the database under a different name than your own login name, or with an `/AuthenticationType` that requires some different user name.

`/AuthenticationPassword:password`

Specifies a password that `mbutility` should use when accessing the database through SQL Server. This must be specified if you wish to access the database under a different name than your own login name, or with an `/AuthenticationType` that requires a password.

`/OrganizationName:name`

The name that should be used in the user's organization list. This name is case-insensitive.

`/DataBaseServer:servername`

The name of the server where the database resides.

`/DataBaseName:name`

The actual name of the database that you want to add.

You must specify all three of `/OrganizationName`, `/DataBaseServer` and `/DataBaseName`.

`+AllUsers`

Adds the organization to the special `AllUsers` list rather than to the list associated with an individual user. Organizations in the `AllUsers` list are available to all users (although security roles still control what the user can or can't do with the database). The `AllUsers` list may be particularly helpful for ClickOnce users. Organizations may be deleted from the `AllUsers` list by using the `DeleteOrganization` verb. The current `AllUsers` list may be displayed with the `ListOrganizations` verb.

If both `+AllUsers` and `+SetDefault` are specified in the command line, *the organization's database becomes the default for all users at your site*. This may be surprising for users whose default is some other database, so you should think carefully before doing this. If a user's default database is different from the `AllUsers` default, the Pick Organization window will be opened the next time that user starts MainBoss, so that the user can see which database is now the default.

Deleting Organizations from the Registry

The `DeleteOrganization` verb deletes an organization from the list of organizations known to a particular user. Deleting an organization does not delete the associated database; it just removes the organization from the list of organizations known to a particular user.

The format of the command is

```
            mbutility DeleteOrganization option option ...
```

where the options are:

`/OrganizationName:name`
: The name associated with the organization in the user's organization list.

`+AllUsers`
: Deletes the organization name from the `AllUsers` list rather than a single user's list. The `AllUsers` list is a special list of organizations which are made available to all users. (Security roles still control what a particular user can or can't do with the database associated with the organization.) Organizations may be added to the `AllUsers` list with `AddOrganization`. The current `AllUsers` list can be displayed with `ListOrganizations`.

Backing Up Your Database with MBUtility

The `Backup` verb of MBUtility lets you back up a MainBoss maintenance organization database. The format of the command is

```
            mbutility Backup option option ...
```

where the options are:

`/BackupFile:filename`
: The name to be used for the backup file. This file should not already exist. Traditionally, SQL Server backup files end in "`.bak`", so

```
            /bu:"c:\SomeFolder\SomeFile.bak"
```

tells MBUtility to save your database's data in a file with the given name.

`/OrganizationName:name`
: The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
: The name of the server that holds the MainBoss database.

`/DataBaseName:name`
: The name of the MainBoss database.

The MBUtility `Backup` verb is subject to all the conditions that affect the behavior of **Administration** | **Backups** within MainBoss itself. For more, see *Backups within MainBoss* on page 50 of this guide.

Backups done with MBUtility are *not* added to the list in **Administration** | **Backups**.

Exporting and Importing Customizations

MainBoss users can customize table viewers, editors and reports by selecting a set of options and giving a name to the result. For example, if a user frequently wants an inventory report with a certain set of grouping and sorting options, with specific pieces of information included in the report, the user can set up all the options, then give the result a name like `My Standard Inventory Report`.

Actions within `mbutility` make it possible to transfer customizations from one user to another. For example, if user A comes up with a useful set of customizations, `mbutility` makes it easy for A to send the customizations to user B. This can also be done with operations inside MainBoss itself, but `mbutility` makes it possible to perform the transfer directly from command lines.

The `ExportCustomization` verb exports all of a user's customizations into a file. It must be executed while logged in to Windows under the user's login name. The format of the command is

```
mbutility ExportCustomization option option option
```

where the options are:

`/Customizations:file`
The name of a file where the customizations should be stored.

`/OrganizationName:name`
The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
The name of the server that holds the MainBoss database.

`/DataBaseName:name`
The name of the MainBoss database.

The `ImportCustomization` verb imports a set of customizations that were exported with `ExportCustomization`. Typically, you would login to a different Windows login name and import customizations created by a different user. However, it may also be useful to import customizations that you yourself exported. For example, if you want to experiment with new customizations, you might use `ExportCustomization` first to save your existing customizations; then, if you don't like your new customizations, you can use `ImportCustomization` to import all your old customizations.

`ImportCustomization` takes the same options as `ExportCustomization`. In this case, however, `/Customizations:file` is used as input rather than output.

Working with Active Directory

Several `mbutility` commands are designed to work with the Windows Active Directory.

```
mbutility AddContactfromactivedirectory options
```

creates one or more contact records in a MainBoss database using information from Active Directory. Note that this process can take a very long time, since Active Directory lookups tend to be slow.

The options for the command are:

`/OrganizationName:name`
: The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
: The name of the server that holds the MainBoss database.

`/DataBaseName:name`
: The name of the MainBoss database.

`/Emailaddresses:address,address,address,...`
: A list of email addresses. Each corresponds to an email address in your Active Directory. `mbutility` will look up each address in the Active Directory and create a record in the MainBoss **Contacts** table using Active Directory information.

The `mbutility` command also has `AddRequestorfromactivedirectory`. This has the same options as the previous command, but it creates records in the **Requestors** table. Typically, this involves creating a contact record first, then a requestor record referring to that contact record.

The `UpdateContactsfromactivedirectory` command will scan your Active Directory and create or delete contact records based on what Active Directory contains. The options are:

`/OrganizationName:name`
: The name of the maintenance organization (associated with your MainBoss database). This name is case-insensitive; for example, `MainBoss` is the same as `mainboss`.

`/DataBaseServer:servername`
: The name of the server that holds the MainBoss database.

`/DataBaseName:name`
: The name of the MainBoss database.

`/Emailaddresses:address,address,address,...`
: A list of email addresses. Each corresponds to an email address to be checked against your Active Directory. `mbutility` will locate the MainBoiss contact records associated with these addresses, and will update the contact information based on the contact information in Active Directory.

`/EmailAddressesPattern:pattern`

A regular expression pattern representing one or more email addresses. `mbutility` will check all contact records whose email addresses match the given pattern and will update those records using information from Active Directory.

`/ExcludedEmailAddressesPattern:pattern`

Tells `mbutility` not to update contact records whose email addresses match the given pattern, even if they would otherwise be updated.

`/ExcludeEmailAddresses:address,address,...`

Tells `mbutility` not to update contact records with the given email addresses, even if they would otherwise be updated.

`+FromEmailAddresses`

Uses the primary email addresses in the selected contact records as well as their "**AD Reference**" fields in order to locate Active Directory matches.

`+DeleteContacts`

If a selected contact record has no corresponding entry in Active Directory, `mbutility` will delete the record. In other words, this removes the contact information for selected people who are no longer in Active Directory.

`+RemoveActiveDirectoryReference`

Clears the "**AD Reference**" field for any selected contact record that does not have a valid Active Directory reference.

`+PreservePrimaryEmailAddress`

Preserves the primary email address for selected contact records if the field isn't blank. Otherwise, the contact record's primary email address field will be filled in with the primary email address from the Active Directory.

`+UpdateAll`

Applies to all contact records that have values in their "**AD Reference**" fields. `mbutility` will update the fields in each such record, using information from Active Directory. Note that `+UpdateAll` and `+FromEmailAddresses` may be used together.

`+Verbose`

Provides a full explanation of every change that is made. We strongly recommend that you use `+Verbose` the first few times you use this operation and that you check the output to make sure that the operation is working the way you expect.

Appendix A: Setting Up SQL Server

Note: All web addresses given in this appendix are subject to change without notice. If Microsoft does change a web address, go to `microsoft.com` and search for the appropriate page.

MainBoss needs to use one of the versions of SQL Server specified in *Microsoft SQL Server* on page 2. If you already have this software installed at your site, MainBoss can use your existing installation. If you do not already have the software, you can purchase SQL Server from Microsoft or download the free version of SQL Server called SQL Express.

Before you look into obtaining any version of SQL Server, check to see if your company already has one installed. Many organizations may already have SQL Server in connection with some other software package.

There are several important options that should be specified when you install SQL Server. Therefore, make sure to follow the installation instructions given later in this appendix.

Server Hardware Requirements: The machine where SQL Server is running should meet the hardware requirements specified by Microsoft for the version of SQL Server you will be using. If you will be running MainBoss on the same computer, the machine should also meet the hardware requirements for MainBoss (stated in the main body of this guide). Since MainBoss and SQL Server will run simultaneously, the computer should have at least the amount of memory required by SQL Server *plus* the amount required by MainBoss.

The speed and available memory of the computer where SQL Server runs has a major effect on the overall speed of MainBoss. The effect is far greater than the speed of the computers where people actually run MainBoss.

Authentication within SQL Server: SQL Server offers two types of user authorization: SQL Authentication or Integrated Authentication. MainBoss requires Integrated Authentication. Mixed mode (accepting both authentication types) will work with MainBoss, but mixed mode is offered only for backward compatibility and Microsoft has deprecated it.

Instances

You can have multiple independent copies of SQL Server running simultaneously on the same computer. Each of these is called an *instance* of SQL Server. Each instance maintains its own separate set of database files.

If you already have SQL Server running at your site, it presumably manages a set of databases. Your IT department may prefer to keep your MainBoss database(s) separate from databases for other software packages. You can do that by creating a new instance of the SQL Server. This can be set up so that it only deals with MainBoss databases.

Different instances of SQL Server are distinguished by having different *instance names*. This is just an identifying name. One instance can have its name blank, in which case it's called the *default instance*. If someone creates a SQL Server database on a computer without specifying an instance name, the database is created and managed by the default instance.

By default, the installation procedures for full versions of SQL Server set up an instance with its name blank. However, for SQL Express, the default installation procedure specifies the instance name `SQLEXPRESS`.

Note: Don't create multiple instances of SQL Server unless you have a good reason. Running multiple instances introduces complexity; it can lead to confusion, inconsistency and human error in how SQL Server is managed. In addition, the computer will probably run more slowly.

SQL Server Installation

The default installation procedures for full versions of SQL Server are designed to let other computers connect with the Server computer. Therefore, their default options will generally work well with MainBoss.

However, the default installation procedure for SQL Express assumes that it will only be used on a single computer, without users on other computers trying to connect with SQL Server. If you intend to have users on other machines connect with SQL Express, you must change a number of default options during the installation. You must also set several configuration options *after* SQL Express is installed, as discussed later in this section. For more information, see *SQL Express Considerations* on page 73.

Overall, we believe customers will find it easier to use one of the full versions of SQL Server than the free SQL Express—full versions are simpler to install and to use (e.g. to back up the database). However, SQL Express is available at no charge and can be used successfully with MainBoss, especially by small maintenance departments.

If you will be installing SQL Server, we recommend that you read Microsoft's background material provided at

https://docs.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-

In particular, make sure you comply with the specified hardware and software requirements.

Default Backup Folder: SQL Server 2008 and R2 let you specify the default folder to be used to hold backup files. We recommend that you set the default folder to be some device *other* than the disk drive that holds the actual MainBoss database. If your backups go to the same disk that holds your original database, a hardware error on that disk may kill both your database and the backups.

Backups with SQL Server

If you have the full SQL Server, the easiest way to implement backups is to use the automated facilities called *maintenance plans*. To set up a maintenance plan, follow these steps:

1. Start SQL Server Management Studio and connect to the SQL Server instance that manages the MainBoss database.
2. In the left-hand panel, expand **Management**.
3. Click **Maintenance Plans**.
4. In the right-hand panel, right-click. In the resulting menu, click **Maintenance Plan Wizard**.
5. The Maintenance Plan Wizard will walk you through the process of setting up automatic backup procedures.

We do not recommend using the default save folder for the backup file, unless you set the default folder as discussed in the preceding section of this guide. SQL Server's usual default backup folder makes backup files difficult to find and work with, especially on some versions of Windows. Instead, we recommend that you save backup files in some folder that you'll remember and find easily. Also, backup files should be copied to some medium that can be removed from the premises; it's important to keep a recent backup off-site in case of fire.

Some commercial backup software packages can work with SQL Server so that SQL Server backups are included when the rest of your computer is backed up. For more information, see the documentation for your backup software.

A formal backup using SQL Server Management Studio is better than just saving the raw disk file(s) containing the database, since a raw save may leave a small amount of data unusable. However, a raw save every day is better than a formal save once a week—losing a week's worth of work is worse than losing a transaction or two.

Note: From time to time, you should check the SQL Server logs to make sure that backups really are taking place successfully. Thinkage has encountered situations where the backup software reports no problems, but where the SQL Server logs show that the database backup failed.

If you do backups directly through SQL Server Management Studio, information about the backup will *not* appear in MainBoss's internal database history (**Administration** | **Database Management**). Furthermore, the backup will not appear in MainBoss's list of known backups (**Administration** | **Backups**).

Using SQL Server to Restore a Database from Backup

To restore a database from a backup file, follow these steps:

1. Start SQL Server Management Studio and connect to the SQL Server instance that manages the MainBoss database.
2. In the left-hand panel, expand **Databases**.

3. Right-click on the entry for the database you want to restore, then click **Tasks**, then click **Restore**, then click **Database**.
4. SQL Server opens a window displaying your existing backup files. In the list of available backups, click the one you want to restore.
5. Click **OK**.

The restoration process will overwrite any existing contents of the database. Therefore, you should only restore a database if the current contents are no longer usable.

SQL Express Considerations

SQL Express is a free version of SQL Server. It is freely available from Microsoft. For up-to-date information on downloading SQL Express from Microsoft, go to `http://www.mainboss.com/info/microsoft.htm`

SQL Express will provide acceptable performance for small sites. Larger sites should buy one of the larger versions of SQL Server.

We should note that SQL Express lacks some features of paid-for versions of SQL Server. For example, SQL Express does not have automatic maintenance plans, Microsoft's facilities that simplify automatic backups and integrity checks.

The rest of this appendix describes considerations for installing and working with SQL Express.

Installing SQL Express

The following description assumes that you do not have any instance of SQL Server installed on your computer. To determine whether you do have such an installation, go to the Windows Control Panel and check "Programs and Features".

Microsoft recommends that SQL Express be installed with a limited permission set and/or as a limited user. For information on this process, see `https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-windows-service-accounts-and-permissions`

If you ever intend to have multiple MainBoss users, we offer the following suggestions as options to choose during installation. Your site might decide to make different choices, but our suggestions should help typical users find their way through the many possibilities.

Unless otherwise stated below, we recommend that you accept the default options in each phase of the installation.

- In the **Instance Configuration** phase of the installation, specify a suitable name for this instance of the server. The default is `SQLEXPRESS`; however, if you are only going to use this instance of the server for MainBoss, you might specify the name `MAINBOSS` to make it clear what the instance is being used for.

- In the **Database Engine Configuration** phase, in the **Data Directories** section, you have the ability to specify a backup directory: the default location for backup files. Ideally, this should be a directory on a different computer than the one holding the original database files—you don't want your originals and your backups to be on the same computer, because if something happens to that computer, you lose everything.

  Also in the **Database Engine Configuration** phase, in the **Account Provisioning** section, you can specify who will have administration permissions on this instance of SQL Server. By default, the list will only contain your own name (i.e. the name of the person installing SQL Express). If someone else will be in charge of adding new users to MainBoss, that person will also need SQL Server administration permissions and should be added to the list.

- In the **Server Configuration** phase, set the SQL Server Browser's startup type to `Automatic`.

The options above constitute a standard configuration for use with MainBoss. They match Microsoft's recommendations for SQL Server. If you make different configuration choices, you may find you have to make changes in future. Also, if you run into trouble, MainBoss Technical Support may recommend that you reset your SQL Server options as listed above.

SQL Express Installation Notes: After installing SQL Express, you must enable TCP/IP connections. This process is described later in this guide in the section *SQL Server Configuration* on page 83.

If you intend to connect with SQL Express from other computers, you must also adjust the Server computer's firewall to allow remote users to connect with the SQL Server and the SQL Server Browser. This process is explained in *Adjusting the Firewall* on page 85.

Finally, you may have to adjust your local security settings to allow access to shared files. This process is explained in *Local Security Settings* on page 88.

Backups with SQL Express

Maintenance plans aren't available with SQL Express. We recommend that you use the backup facilities within MainBoss.

Important: By default, SQL Server configures itself so that backup files are written to the same folder as the databases themselves. **This offers no protection against hardware errors**—if you get a disk crash on the disk that holds the databases, you'll lose the backup files too. Therefore, you should configure SQL Server so that backup files are written to a completely different device, preferably on a completely different computer.

Ideally, you should make copies of your backup files and keep those copies off your premises. That way you don't lose data if there's a fire or some other disaster that affects your entire site.

SQL Server Maintenance

From time to time, you should clean up your SQL Server database. Clean-up operations can include reorganizing the database index, rebuilding the index from scratch, and various other actions.

Note: SQL Server uses various working files in its processes. It keeps its working files around until the main database files get backed up, just in case the computer crashes and SQL Server has to try to reconstruct everything that's happened since the last backup.

For this reason, you shouldn't back up your database just by making a simple copy of the database file. You must use some technique which tells SQL Server, "We've made a good backup, so you can get rid of the working files." Otherwise, the working files will continue to grow and grow. Most commercial backup software has facilities that can work with SQL Server databases in the appropriate way.

If you have a full SQL Server (as opposed to SQL Server Express), you can arrange clean-up operations through SQL Server Management Studio. To do so, follow these steps:

1. Login to an account with Windows Administrator privileges.
2. Start SQL Server Management Studio. (On Windows 7 and Vista, you should use **Run as Administrator**.)
3. In the left-hand panel, expand the entry for **Management**.
4. Right-click the entry for **Maintenance Plans**, and click **Maintenance Plan Wizard**.

The wizard will then walk you through the process of creating a maintenance plan that performs various operations automatically. During the process, you can choose what operations will be performed. As a minimum, we recommend the following:

```
Check Database Integrity
Reorganize Index
Rebuild Index
Clean Up History
Maintenance Cleanup Task
```

(You can also include a backup operation, although you should already be backing up your database with a different maintenance plan—you don't need to clean your database as often as you need to back it up.)

Once you've used the wizard to create the maintenance plan, you can execute the operations by right-clicking the entry for the maintenance plan, then clicking **Execute**.

If you ever have a system crash or power failure on the server computer, you should make a point of cleaning up your database as soon as possible afterward.

If you have SQL Express, the associated Management Studio does not allow you to make maintenance plans.

SQL Server and MainBoss Service

MainBoss Service is a Windows service which handles MainBoss's incoming and outgoing email messages. It can run on any computer in your network which can access the computer where SQL Server runs. For more information, see *MainBoss Service* on page 24.

MainBoss Service runs under the `Network Service` login name on the computer where the service is installed. When such services interact with other computers, they do so under a name consisting of the computer name plus a dollar sign `$`. For example, suppose MainBoss Service has been installed on a computer named `MSCOMPUTER`. Then it interacts with other computers on the network using the login name `MSCOMPUTER$`. In particular, when this MainBoss Service makes use of SQL Server, it will do so under the name `MSCOMPUTER$`.

MainBoss Service must be installed by someone with SQL Server Administration privileges on the instance of SQL Server that manages the MainBoss database. The installation process makes use of these privileges to set up security permissions that will allow MainBoss Service to access the MainBoss database. In particular, the installation process adds an appropriate login name to SQL Server's list of recognized names. Using our previous example, this would mean adding `MSCOMPUTER$` to SQL Server's list of authorized login names.

In our experience, some SQL Server administrators are unaccustomed to working with such names. If they notice a name like `MSCOMPUTER$` in the list of authorized login names, they may delete it because it looks unusual. Deleting the name will prevent MainBoss Service from accessing the MainBoss database; the service will stop processing incoming and outgoing email messages and will write error messages to the MainBoss Service log until the problem is corrected.

Appendix B: Troubleshooting

This appendix offers some help for determining whether SQL Server and other software are set up properly for use with MainBoss. The appendix also describes tests for checking whether a particular user can access the SQL Server databases. Finally, the appendix provides some tips on how to approach non-standard MainBoss configurations if your IT department insists on deviating from MainBoss's recommended set-up.

Note: This appendix uses "the Server machine" to refer to the computer where SQL Server is running and where the MainBoss database is stored. We assume that the MainBoss database already exists, created in accordance with the instructions in the main body of this guide. "The Remote machine" is a different computer from which you want to access the MainBoss database; this computer should be on your local area network (LAN).

General Advice

Many things might go wrong when trying to use SQL Server. Unfortunately, Microsoft's diagnostic facilities aren't always helpful. If an error condition occurs with SQL Server, MainBoss can only display the diagnostic message that SQL Server or Windows provides, and this might not tell you much.

If you get a diagnostic message that doesn't tell you enough, use Google to search `microsoft.com` for references to the *Details* section of message. (If the Details section is blank, use the main part of the message.) For example, if the Details section of the message is `"Some text"`, you would type the following into Google's search field:

```
site:microsoft.com "Some text"
```

The `site:microsoft.com` tells Google that you only want responses from Microsoft's web site. Putting quotes around `"Some text"` means that you search for that exact string.

Note that if the error text contains specific references to names of computers, users, etc., you must remove them before you do the search and only put quote marks around each separate section. Also, if the message is very long, you can just extract important sections. For example, if you get the message

```
Login failed for user 'JSMITH'. The user is not associated
with a trusted SQL Server connection.
Unable to create the base database session
```

you could write up the Google search as

```
site:microsoft.com "Login failed for user" "trusted SQL Server connection"
```

Several of the sections in this appendix describe tests you can try if something isn't working. Different tests are certainly possible, but the ones we describe provide a productive route for checking potential sources of difficulty. All of these tests use standard Microsoft software. If the tests fail somewhere, you can hope the software provides an error message that explains what went wrong. You can then fix the problem and try again. Once the test works with Microsoft software, you can try again with MainBoss itself.

The tests described in this section should be carried out in the order given. The purpose of each test is to detect problems in your set-up of Windows, your network, or SQL Server and then to determine where the problem lies. It’s important to eliminate the possibility of such problems before contacting MainBoss Support—don’t mistake a problem in Windows for a problem in MainBoss.

If you have problems with Windows, your network, or SQL Server, contact Microsoft or your software service provider.

File-Sharing

Before other computers can connect with the Server machine, the Server’s firewall must be set up to allow file-sharing.

If you have a commercial firewall product (e.g. one that comes as part of an antivirus package), consult the software’s documentation for how to allow file-sharing. If you’re using the standard Windows firewall, follow these steps:

1. On Windows 7, Vista or Server 2008:
    - In the Control Panel, click **Windows Firewall**.
    - In the resulting window, click **Change settings**.
    - A window opens where you can make firewall adjustments. Go to the **Exceptions** section of this window.
2. On later versions of Windows:
    - In the Control Panel, use the search field to find “**Windows Firewall**”.
    - Click “Allow an app or feature through Windows Firewall”.
3. In the list, go to `File and Printer Sharing`. If the accompanying checkbox isn’t already marked, checkmark it.
4. Click **OK** to exit.

This step is required for all the other troubleshooting discussed later in this appendix.

Note also that MainBoss users need read permission on whatever folder holds the MainBoss software that they’re supposed to use. If you install the software on individual computers, make sure that on each computer, every person running MainBoss on that computer has read permission on the MainBoss software folder. If you install the software at a shared network location, make sure that every person using that software to run MainBoss has read permission on the shared folder.

Testing for the Server's Machine Presence

To check connections between the Server and Remote computers, the most basic test is a *ping*:

1. On the Remote computer, open a Command Prompt window.
2. Type `ping`, followed by a space, followed by the name of the Server computer. For example, if the Server is called `OurServer`, you'd type

```
ping OurServer
```

Wait for results. If the connection can't be made, you'll get an error message or a time-out message (i.e. the `ping` command waited a significant length of time but never got an answer). If the connection can be made, you'll get a message saying that the Server computer replied and providing other statistics.

If `ping` fails, check the documentation on your network software to see what's gone wrong.

There are two types of pings: `ping -4` and `ping -6` (which test for IP4 and IP6, respectively). SQL Server generally uses IP4, so you usually want to use `ping -4`.

Testing That a User Can Connect with the Server Machine

This test determines if a particular user on a particular Remote machine can connect with the Server machine, and that the user has appropriate permissions to access the machine.

The user's login name and password must be the same on the Remote machine as on the Server machine. This will automatically be true if you're using domains. If you aren't using domains, have the user attempt to login to the Server machine using his/her login name and password. If this doesn't work, the login name and/or password must be changed so that they're the same on all relevant systems.

Note: If you aren't using domains, users should be warned that if they change their passwords on one computer, they should change the password on all other computers too. This is especially important on the Server machine.

It's important to note that SQL Server has its own security permissions that are separate from Windows permissions. For example, a disk may have general read/write permissions, allowing any user to read or write any file on the disk; still, SQL Server may only permit access to a limited set of users based on login name and password. If you can't connect to SQL Server, here's a way to make sure that a particular user on a Remote machine has login and password set up correctly.

Before you do this, you have to make sure that you don't have any existing connections to the Server computer. To do that, follow these steps:

1. On the Remote computer (where you'll be doing this test), open a Command Prompt window.
2. Type `net use` (this displays any existing connections).
3. If there are no connections listed for the Server computer, you don't have to do anything; just close the Command Prompt window.
4. Otherwise, you have to delete all existing connections to the Server computer.
    - First, write down the information shown about connections to the Server computer. You'll need this information later in order to re-establish the connections. (You can save the output of `net use` in a file by using a command line of the form `net use >file`).
    - If a connection has a letter drive shown in the Local column (e.g. `Z:`), type a line of the form

      ```
      net use Z: /delete
      ```

      where you replace `Z:` with the letter shown.
    - If a connection doesn't have anything in the Local column, it will have a name in the Remote column of the form `\\machinename\folder`. Type

      ```
      net use \\machinename\folder /delete
      ```

      where you replace `\\machinename\folder` with the actual names shown.
    - Repeat the above for every connection to the Server computer. You do not have to delete connections to other computers.
    - When finished, it's useful to leave the Command Prompt window open, since you may need it again later.

Note: The connections that you delete may be needed in order for other software to work. If so, you must re-establish the connections after you've done the test described in this section. We'll discuss re-connections below.

Once you've got rid of the connections, do the following test:

1. On the Server computer, use Windows Explorer to share a test folder with the rest of the network:

- In Windows Explorer, create a new folder anywhere on the Server computer's disk drive(s).
- On Windows 8:
    - Right-click on the folder.
    - In the resulting menu, click **Share**, then **Stop Sharing**.

    - Windows will ask if you really want to remove all sharing permissions. Click **Stop Sharing**.
    - You’ll return to Windows Explorer. Right-click on the folder again.
    - In the resulting menu, click **Share**, then **Specific people**.
    - Click the drop-down arrow for the field labeled **Type a name and then click Add, or click the arrow to find someone**.
    - In the resulting drop-down menu, click **Find people**.

- On Windows 7 or Vista:

    - In Explorer’s **Tools** menu, click **Folder Options**.
    - In the **View** section of the resulting window, go down to the bottom of the list of settings and turn off **Use Sharing Wizard** (on Windows 7 or Vista).
    - Click **OK**.
    - Right-click on the folder, then click **Share**.
    - In the **Sharing** section of the resulting window, click **Advanced Sharing**.
    - In the resulting window, checkmark **Share this folder**.
    - Click **Permissions**.
    - The resulting window shows who can access the new folder. Remove all existing permissions (if any) by clicking each name in the list, then clicking **Remove**.
    - Once you’ve removed all existing permissions, click **Add**.

- Under **Enter the object names to select**, enter the login name of the person on the remote computer whose connectivity you’re testing. (This should also be a login name on the Server computer.)
- Click **Check Names**. Normally, the name will be recognized as a known login name; however, you may be asked to resolve ambiguities (if, for example, another login name starts with the same set of characters).
- When you have settled on the correct login name, click **OK**.
- When you return to the previous window, checkmark **Full Control** under the **Allow** column (Windows 7 or Vista) or click the drop-down arrow associated with the new user and choose **Read/Write** (Windows 8).
- Click **Share** (Windows 8) then keep clicking **OK** until all windows close.

2. On the Remote computer, have the user log in using his/her login name and password.
3. On the Remote computer, use Windows Explorer to connect to the shared folder.
4. On the Remote computer, attempt to create a text file in the shared folder. Add text to the file (e.g. using Notepad) and save the file.
5. On the Remote computer, attempt to delete the file you just created.

If you can do all this and you are never asked to enter a login name and password during the process, the user's network permissions are set up correctly. If you can't do one of these steps or if you have to type in a login name and password before one of the steps works, you don't have the right permissions. See *Local Security Settings* on page 88 for suggestions of how to deal with the problem.

Once you've verified that the Remote user can pass these tests, the person who created the original (shared) folder on the Server machine can delete the folder.

Again, we emphasize that setting up a domain will avoid a lot of headaches. Microsoft recommends the use of domains with SQL Server.

If you had to remove any connections before testing the file-sharing, you can reconnect them as follows:

1. On the Remote computer, open a Command Prompt window (or return to the one you already had open).
2. If the original connection had a letter drive (e.g. `Z:`), type the following:

   ```
   net use Z: \\machinename\folder
   ```

   where you replace `Z:` with the letter drive it had previously and replace `\\machinename\folder` with the previous value. You may be asked to enter a login name and password.
3. If the original connection had no letter drive, type

   ```
   net use \\machinename\folder
   ```

   where you replace `\\machinename\folder` with the previous value. Again, you may be asked to enter a login name and password.
4. Repeat the above for every previous connection to the Server computer.

Comparing the DNS Name and Active Directory Name

It is possible for the DNS name of a system to get out of synch with the Active Directory name. If the names get out of synch, all kinds of anomalous problems may occur. Therefore, it's important to make sure these names refer to the same thing. To do this, follow these steps (on both the server computer and the remote computer where you're having trouble):

1. Open a Command Prompt window.
2. Type the following:

   ```
   ipconfig
   ```

   The system will display a list of information, including the IP address of the computer.
3. Type the following:

```
nslookup
```

The system will display some information, then type > to prompt for more input.

4. Type the name of the current computer (then press **<Enter>**). The computer will type the IP address of the computer.
5. Type `exit` to quit the `nslookup` program.

Both `ipconfig` and `nslookup` should display the same IP address for the computer. If not, the names are out of synch and you'll have to reset one or the other. For information on resetting these names, see your Windows documentation.

SQL Server Configuration

As noted in *SQL Server Installation* on page 71, the default installation procedures for SQL Server set things up so that SQL Server allows access to users on other computers via TCP/IP. However, they may have been disabled after installation. (Also some older versions of SQL Server have TCP/IP disabled by default.) Therefore, you may need to enable TCP/IP connections. To do so, follow these steps:

1. In the Windows **Start** menu, locate the entry for your version of SQL Server. Under this entry, locate **Configuration Tools** and then **SQL Server Configuration Manager**. This opens a window where you can configure your SQL Server. (In Windows 8, just search for **SQL Server Configuration Manager** from the desktop.)

If you can't find an entry for **SQL Server Configuration Manager**, you'll have to add the software. See "Maintenance Installs" below.

2. In the left-hand side of the window, expand the entry for SQL Server Network Configuration.
3. In the resulting expansion, click **Protocols for MAINBOSS**. (This will have a different name if you chose a different instance name.)
4. If MainBoss will be used on a network, with or without a domain, right-click on **TCP/IP** in the right-hand side of the window and click **Enable** (if this isn't already enabled). You do not have to do this if you'll only be using MainBoss on a single computer.

   A message will appear saying that your changes will only take effect once you stop and restart the service. The steps below do exactly that.

The list of protocols also offers the possibility of enabling SQL Server access through named pipes. However, Microsoft recommends that you don't use named pipes across a network. See below for more on named pipes.

5. In the left-hand side of the window, click the entry for SQL Server Services. You should see an entry for **SQL Server (MAINBOSS)**. (If you used a different instance name than **MAINBOSS**, that name will be displayed instead.) Right-click on this entry, then click

STOP. (This will actually stop both instances displayed in the right-hand side of the window.)

6. Right-click on **SQL Server (MAINBOSS)** and click **START**.
7. Right-click on **SQL Server Browser** and click **Properties**. This opens a window where you can set the browser's properties.
8. In the **Service** section of the properties window, click **Start Mode**.
9. Drop down the associated arrow (at the end of the line) and click **Automatic**.
10. Click the window's **Apply** button.
11. In the **Log On** section of the same properties window, click **Start** to start the browser again.
12. Click **OK** to close the window.

Named Pipes: As noted above, you have the option of allowing SQL Server to use named pipes in addition to TCP/IP. We strongly recommend that you disable named pipes. There are several reasons.

- Each named pipe counts against the maximum number of sessions that Windows allows, while TCP/IP connections do not. Therefore, using named pipes decreases the number of people who can use the Windows system for other purposes.
- The default permission structure associated with named pipes essentially means that Windows Administrators can access SQL Server but nobody else can. It's possible to change the structure, but this is seldom done.
- If you *do* enable named pipes (e.g. because some other software using SQL Server requires them), you'll have difficulties debugging any TCP/IP problems that might arise. The reason is that the .NET framework (which underlies MainBoss) attempts to contact SQL Server first through shared memory; this won't work if SQL Server is running on a different computer. Next, .NET tries TCP/IP, which is the preferred connection method...but if this fails for any reason, .NET immediately goes on to try named pipes. You will *not* receive any error message explaining what happened to TCP/IP. If named pipes work, they'll count as a Windows session (as discussed above), and if they don't work, you'll get an error message related to named pipes, rather than one telling you the problem with TCP/IP. In other words, named pipes simply interfere with your ability to get TCP/IP working properly.

As this discussion shows, we see no good reason for using named pipes and a number of reasons not to. Avoid them if possible.

Maintenance Installs: When your SQL Server was installed, it's possible that SQL Server Configuration Manager was omitted—its installation is optional. If you're missing the configuration manager, you'll have to do a *maintenance install* to add the software.

To do this, start the SQL Server or SQL Express installation again. Tell the installation procedure that you want to change the installation, then follow the suggestions given in *Installing SQL Express* on page 73.

Starting the SQL Server Browser: The SQL Server browser makes it possible for other computers to detect SQL Server on the server system. If the browser is not running, instances of SQL Server on this computer will not appear in the drop-down list for "**Database Server**" when you are specifying a maintenance organization. (See, for example, *Creating a Maintenance Organization* on page 7.)

By default, the browser is installed when you install the rest of SQL Server. For full versions of SQL Server, the browser is also set up to start automatically. However, for SQL Express, the browser is *not* set up to start automatically. To set it up that way, follow these steps:

1. In the Windows **Start** menu, locate the entry for the version of SQL Server you are using. Under this entry, locate **Configuration Tools** and then **SQL Server Configuration Manager**. This opens a window where you can configure your SQL Server.
2. In the left-hand panel, click the entry for SQL Server Services.
3. In the right-hand panel, examine the entry for **SQL Server Browser**. If the state is `Running`, the browser is already configured to be active. You can quit and skip the remaining steps below.
4. If the state is `Stopped`, you must set up the browser for automatic start. Right-click the entry for **SQL Server Browser**, then click **Properties**.
5. In the resulting window, go to the **Service** section.
6. Click the entry for **Start Mode**. Click the value associated with this entry; this opens a drop-down list. In the list, click `Automatic`.
7. Click **OK**.
8. When you return to the previous window, right-click the entry for **SQL Server Browser**. In the resulting menu, click **Start**.
9. Exit the Configuration Manager.

Adjusting the Firewall

In order for other computers to access SQL Server on the Server machine, you must configure the Server machine's firewall to allow such access. (If SQL Server is already in use at your site, the Server machine is probably configured appropriately all ready.)

> If you're running an anti-virus program, it has probably installed its own firewall. Check the program's documentation for how to grant access to SQL Server.

Granting Access to a Program through the Firewall

If you have commercial firewall software (typically part of an antivirus package), follow the software's instructions to let SQL Server and the SQL Server browser through the firewall (as discussed in the previous section). If you use the built-in Microsoft firewall, follow these steps.

1. On Windows Vista or Server 2008:
    - In the Control Panel, click **Windows Firewall**.
    - In the resulting window, click **Change settings**.
    - In the resulting window, go to the **Exceptions** section.
2. On Windows 7 or 8:
    - In the Control Panel, click **System and Security**.
    - In the resulting window, click **Windows Firewall**.
    - In the side panel, click **Allow an app or feature through Windows Firewall**.
3. Check in the list to see if `sqlbrowser.exe` and `sqlservr.exe` are already there and checkmarked. If so:
    - Double-click on `sqlbrowser.exe`. A window will open to show you the full pathname of the file. Check that the pathname refers to the SQL Server you will be using. (This is necessary in case you have multiple versions of SQL Server on your computer, or if the entry for `sqlbrowser.exe` refers to the software for some previous version of SQL Server.) If so, click **OK** to exit.
    - Repeat the above procedure for `sqlservr.exe`.
4. If `sqlbrowser.exe` and `sqlservr.exe` are both in the list, both checkmarked, and both refer to the correct programs, you're finished—click **OK** to exit.
5. If the correct files are in the list but not checkmarked, checkmark them then click **OK** to exit.
6. If one or both file names are missing, click **Add Program**.
7. In the resulting window, click **Browse**.
8. In the resulting window, find the name of the SQL Server executable file (see the previous section of this guide for how to find this name). Click the file name, then click **Open**.
9. You'll return to the previous window. Click **OK**. The file is added to the list.
10. Repeat steps 7-10 for the SQL Server browser. Click **OK** when you're done.

Scopes: Each program in the Windows Firewall list has an associated *scope* (unrelated to scopes in MainBoss). The scope may be any of the following:

- **Any computer**, in which case the firewall lets the program be used by any computer anywhere (including those on the Internet).
- **My network**, in which case the firewall only lets the program be used by computers in your local area network.
- **Custom list**, in which case the firewall only lets the program be used by a specified list of computers (e.g. ones identified by IP number).

Most sites will wish to restrict MainBoss access to users on your local area network (**My Network**), not to users anywhere on the Internet. If you open up access to all Internet users, you must be sure that you've protected yourself with appropriate security measures.

Your usage requirements may change with time. For example, you may originally restrict MainBoss access to **My network**. Later on, you might decide that you want to be able to connect from outside sites (e.g. when working at home or at a remote job site), in which case you'll have to change your configuration.

To see the scope associated with a program (and possibly to change it), follow these steps:

1. Follow steps 1-3 listed earlier in this section.
2. In the list of exceptions to the firewall settings window, click the entry for `sqlservr.exe`.
3. Click **Properties**.
4. In the resulting window, click **Change Scope**.
5. In the resulting window, check the scope. If it isn't what you want, change the setting.
6. Click **OK** twice to return to the firewall settings window.
7. Repeat steps 2-6 for `sqlbrowser.exe`.

Firewall Off and On

Some versions of Windows incorrectly report that the firewall is off when it's actually on. Specifically, suppose you go to **Windows Firewall** in the Windows control panel. Your computer will respond with a window that tells you whether the firewall is off or on, *as set by the last time you used this window*.

However, it is possible to turn the firewall off/on using other windows. In particular, you can turn it off or on using the Services window (obtained by selecting **Manage** on **Computer** or **My Computer**). If you turn the firewall off or on using the Services window, the other window (accessed through the control panel) doesn't change.

As a result, the firewall may be on, even though the control panel says it's off. Even more confusing, it is sometimes possible for administrators to access the database just fine, but others cannot. We've seen this happen because the firewall was letting named pipes access the MainBoss database; people allowed to use the named pipes could get access while others could not.

If you find yourself in this position, go to the Services window and make sure that the Windows Firewall service is actually in the condition you think it is (stopped or started). You may need to start or stop the service in order to put it into the state you actually want.

Local Security Settings

If you are not using domains and you intend to use MainBoss on multiple machines, you may have to change the security settings on the computer where SQL Server is running. **This is unnecessary if you're using domains.**

In many cases, the security settings will already be correct by default. However, some versions of Windows do not have the correct settings. Also, if you upgrade your version of Windows, the upgrade process retains your old settings, even if they're different from the usual defaults of your new operating system.

To set up the correct settings, follow these steps on the computer where SQL Server is running:

1. Open the Windows control panel.
    - In Windows 7 and 8, click **System and Security**, then **Administrative Tools**.
    - In Windows Vista and Server 2008, click **Administrative Tools** directly from the control panel.
2. In the administrative tools menu, click **Local Security Policy**. This opens a window where you can change the security settings.
3. In the left-hand panel of the window, expand **Security Settings**, then expand **Local Policies**, then click **Security Options**. MainBoss opens a list of security options you can set.
4. Double-click on the entry for **Network access: Sharing and security model for local accounts**. This opens a window where you can set the option.
5. In the drop-down list, choose the entry for **Classic - local users authenticate as themselves**.
6. Keep clicking **OK** until you've closed all the windows.

The alternative to **Classic** is that anyone logging into the computer will end up as a user called `Guest` (provided that `Guest` is enabled which is not true on many systems). Such a user would not be able to use MainBoss unless you authorized `Guest` as a MainBoss user...but then *anyone* could use MainBoss since `Guest` usually requires no password.

Authorizing Users in SQL Server

If you have checkmarked the option **MainBoss manages SQL Security** in the **Defaults for Users** section of **Administration** | **Users**, then MainBoss automatically gives new users appropriate permissions to access the MainBoss database through SQL Server. If you have turned off this option, you must authorize users manually. You might also have to authorize a user manually if someone else (or another program) has de-authorized the user for some reason.

> In order to follow the steps described in this section, you must have SQL Server Administrator permissions on the SQL Server that manages MainBoss.

The following steps describe how to check whether a user has appropriate permissions for accessing the MainBoss database through SQL Server. The steps also describe what to do if permissions for a user have somehow been removed:

1. On the computer where SQL Server is running, start SQL Server Management Studio.
2. When Management Studio asks to specify the "**Server Name**", give the name of the instance of SQL Server that holds the MainBoss database. (This is the same server name you specified when you created the database.)
3. Click **Connect**. Management Studio will open a window showing the configuration of the given instance of SQL Server.
4. In the left-hand panel, expand the entry for **Security**.
5. Under **Security**, click **Logins**. Management Studio will display a list of authorized users. This list may contain user groups as well as the login names of individual users.
6. If the user you want to authorize isn't currently in the list:

   □ Right-click **Logins**.

   □ In the resulting menu, click **New Login...** Management Studio will open a window where you can authorize a new user.

   □ In "**Login name**", enter the login name of the person you want to authorize. This must be a valid login name on the computer where SQL Server is running.

   □ Click **OK**.

7. Once the user's name is on the list of logins, you must check that the user has correct permissions for accessing the MainBoss database. Right-click the name of the user in the right-hand panel, then click **Properties**. MainBoss opens a window providing information about the user.

8. In the left-hand panel of the “Login Properties” window, click **User Mapping**.
9. In the right-hand panel, checkmark the entry for the MainBoss database (if it isn’t already checkmarked). Management Studio will highlight the line and put the user’s login name in the “User” column.
10. The entry in the “Default schema” column should be “`dbo`”. If it isn’t:

    - Click the drop-down button (...) at the end of this line. Management Studio opens a window where you can specify a default schema.
    - Under “Enter the object names to select”, type “`dbo`”.
    - Click **OK**. The original line should now give the name of the MainBoss database, the user’s name, and “`dbo`”.

11. Under “Users mapped to this login”, make sure the name of the MainBoss database is highlighted.
12. Under “Database role membership for:”, checkmark **MainBoss** (if it isn’t already).
13. In the left-hand part of the Login Properties window, click **Status**.
14. Click **Grant** and click **Enabled** (if they aren’t already).
15. Click **OK**.
16. Repeat Steps 6-15 for every user you want to authorize.

Remember that the name you specify must be a valid login name for the current computer. When logging in from other computers, users must have the same name and password as on the computer where SQL Server is running. (This will always be true if you use domains, since the whole point of domains is to let users have the same name and password on multiple machines.)

Spelling: If a particular user can’t access the MainBoss database, always check that the user’s name is spelled correctly in the various places it should appear, e.g. MainBoss’s **Users** table and SQL Server’s **Logins** list.

Groups: SQL Server lets you grant permissions to user groups as well as individuals. Therefore, you might choose to use the above procedure to grant permissions to a “MainBoss User” group. Once you do that, you can add new users to the group and they automatically receive SQL Server permissions on the MainBoss database. However, the **Users** table inside MainBoss *only* allows individual login names, not groups.

Note that you might be tempted to set up SQL Server permissions so that *anyone* can access the MainBoss database. However, MainBoss itself will only work for people explicitly authorized in the MainBoss **Users** table. If you grant SQL Server permissions to all users, you end up with the undesirable situation where some people may be prevented from using MainBoss itself, but could still change the database by hand (e.g. with Microsoft Access) if they wanted to. If someone can access the database, it’s best to make sure that they do it by using MainBoss, not some other piece of software.

Deleting Users: When you add a user to **Administration** | **Users**, MainBoss grants that user permission to connect with SQL Server (if you've turned on the **MainBoss manages SQL Security** option). However, when you delete a user from the **Users** table, MainBoss does *not* delete the user's permission to connect with SQL Server—the user may need "Connect SQL" permission in order to use SQL Server with another software package.

If you wish to completely remove a user's access to SQL Server, remove the user's name from SQL Server's **Logins** list, using SQL Server Management Studio.

Testing Whether a User Has Access to SQL Server and the MainBoss Database

In order to access the MainBoss database from a Remote machine, a user must have appropriate permissions with SQL Server. The following procedure can be used to determine if a user has these permissions. (In order for this to work, the user who performs the test must have administrator privileges, i.e. be part of the `Administrators` group.)

1. Login to the Remote machine in question under the login name you want to test.
2. Open the Windows control panel.
   - In Windows 7 and 8, click **System and Security**, then **Administrative Tools**.
   - In Windows Vista and Server 2008, click **Administrative Tools** directly from the control panel.
3. In the Administrative Tools menu, click **Data Sources (ODBC)** or **ODBC Data Sources**, whichever appears in the list.
4. In the resulting window (ODBC Data Source Administrator), go to the **System DSN** section.
5. Click **Add**.
6. In the list, click **SQL Server** (typically near the bottom of the list), then click **Finish**.
7. Windows opens a box where you can create a new data source for SQL Server. For **Name**, specify anything (e.g. `TEST`). In **Server**, pick the SQL Server where MainBoss resides.
8. Click **Next**, then click **Next** again.
9. Checkmark **Change the default database to:**. In the resulting drop-down list, pick the name of the MainBoss database on the SQL Server system.
10. Click **Next**, then click **Finish**.
11. Click **Test Data Source**.

You should end up with a window saying that the test completed successfully.

Once you get the message that the test completed successfully, you can click **OK** twice. This gets you back to the ODBC System Administrator window. At this point, you can **Remove** the data source that you just added. This doesn't do anything to the MainBoss database, it just removes the entry from the list. Click **OK** to finish up.

Remember, if you get an error during this process, use Google to look up the error message: you may get useful tips about what went wrong.

Testing Database Access with Microsoft Excel

You can test your access to the MainBoss database by using Microsoft Excel. Any user can perform this test, but the Excel software must be installed on your system.

If you have Microsoft Office 2007 or 2010:

1. Login to the Remote machine in question under the login name you want to test.
2. Start Microsoft Excel.
3. In the menu bar, click **Data**.
4. In the ribbon, click **From Other Sources**, then click **From SQL Server**.
5. Excel opens a window for you to specify a SQL Server connection. Under **Server Name**, enter the name of the SQL Server that holds the MainBoss database. Click **Next**.
6. Select the name of the MainBoss database from the drop-down list.
7. The wizard displays a list of tables in the MainBoss database. Since this is just a test, you can select any table; **License** is a good choice since it always contains a few lines of data, but not too much. Select this table.
8. Click **Next**, then click **Finish**.
9. Excel displays a window asking how you want to view the data. Just click **OK**.

If all goes well, Excel will insert your license keys and other data into the current spreadsheet. If this process works, the user has all the correct permissions to access the MainBoss database. If not, error messages from Excel may help you determine what went wrong. You may also find the following article useful:

https://technet.microsoft.com/library/Cc917670

If you have Microsoft Office 2003, the process is similar to the one above. To begin, go to Excel's **Data** menu, click **Import External Data**, and then **Import Data**. In the resulting menu, click **New Source**. From that point onward, the process is essentially the same as above, starting at Step 4.

Testing Database Access with SQL Server Management Studio Express

If you don't have Microsoft Word on a particular system, you can test database access with SQL Server Management Studio Express, available free for download from Microsoft:

https://www.microsoft.com/downloads/details.aspx?familyid=C243A5AE-4BD1-4E3D-94B8
5A0F62BF7796&displaylang=en

Follow these steps:

1. Have a privileged user install SQL Server Management Studio Express on the Remote computer where you wish to test access.
2. Under your own login name, start Management Studio Express.
3. When the software asks you to specify a server name, enter the name of the Server computer, followed by a backslash, followed by the SQL Server instance name where MainBoss is running, e.g. `MYSERVER\MAINBOSS`.
4. Click **Connect**.
5. In the left-hand panel, expand the entry for **Databases**.
6. Expand the entry for the MainBoss database.
7. Expand the entry for **Tables**.
8. Expand the entry for **dbo._DAccessCode**.
9. Expand the entry for **Columns**.

If you can see entries for **Code** and other data fields, SQL Server is allowing you to look at the database. (You don't need to see the actual database values.)

Testing POP3/IMAP4 Permissions for MainBoss Service

The **MainBoss Service** module needs to obtain mail messages from its mailbox using either the POP3 or IMAP4 protocols (techniques for accessing a mailbox). Some sites configure their systems to prevent POP3 and/or IMAP4 from being used. Therefore, it's useful to test whether these protocols are usable on the computer where you've installed MainBoss Service (see *Setting Up MainBoss Service* on page 25).

If you intend to use POP3 or POP3 with SSL, and if you have the Telnet client installed on your system, it's easy to test whether POP3 is enabled. (If you do not have the Telnet client installed, you can install it by opening the **Programs and Features** entry of the Windows control panel. Click **Turn Windows features on or off**, and checkmark **Telnet Client** in the resulting window.)

With Telnet client installed, you can test whether POP3 is enabled by opening a command prompt window and entering

```
telnet SYSTEMNAME 110
```

where `SYSTEMNAME` is the name of the computer that handles your email (e.g. `mailhandler.ourcompany.com`). If the answer begins with the characters `+OK`, then POP3 or POP3 with SSL is enabled. (Type `quit` to end the `telnet` command.)

Similarly, if you intend to use IMAP4 or IMAP4 with SSL, enter the command

```
telnet SYSTEMNAME 143
```

If the answer begins with the character `* OK`, then IMAP4 or IMAP4 with SSL is enabled.

Unfortunately, this approach won't work if you intend to use POP3S or IMAP4S. In that case, the easiest way to check is to try to create a POP3S or IMAP4S mailbox. (This approach will also work to test normal POP3 and IMAP4, with or without SSL.)

Different mail software products have different ways of creating mailboxes. If you have Microsoft Outlook or Outlook Express, you can do the following:

1. In Outlook's **File** menu, click **Add Account**.
2. Click **Next**.
3. In the resulting window, click **Add**.
4. Click either **POP3** or **IMAP** (whichever you're using for MainBoss Service).
5. Click **Next**.
6. If you have configured MainBoss Service to use POP3S or IMAPS (which is recommended), checkmark **Log on using Secure Password Authentication (SPA)**.
7. Continue to fill in information as the wizard requires. Make sure that you use the same port numbers that you specified when configuring MainBoss Service.
8. When you've finished creating the account, send mail to it and see if the mail arrives.
9. When you're finished testing, delete any unneeded test accounts.

As noted, some sites have set up their email systems to prevent POP3 and/or IMAP4, since these protocols can create security holes if used in the wrong context. This has led some sites to ban the protocols entirely, even though they are safe when used in the right context. If your site has banned the protocols, the information in

http://www.mainboss.com/english/products/pop3.htm

may help convince your IT department that the protocols can be allowed for MainBoss.

On Windows 7 and Vista, if you don't have Outlook, you can use the built-in Windows Mail software. The process for adding a new account isn't exactly like the procedure above, but it's similar.

Odd Processing by Outlook/Exchange: Several MainBoss customers have encountered an odd behavior in Outlook and/or Exchange where HTML messages are mistaken for plain text. As a result, Outlook/Exchange tells MainBoss that a message is plain text, when it's really HTML; the result is that MainBoss displays mail messages as raw (hard-to-read) HTML, rather than processing them correctly. For help in fixing this behavior, see our online FAQ article at:

http://www.mainboss.com/english/support/faq/mb30_outlookhtml.htm

Testing that You Can Start and Stop MainBoss Service

As discussed in *Setting Up MainBoss Service* on page 25, you can start and stop MainBoss Service from a computer other than the one where MainBoss Service is installed, provided you have appropriate permissions and the MainBoss Service computer is appropriately configured. As a quick test for checking whether these conditions are met, follow these steps:

1. Open a Command Prompt window.
2. Type `sc`, followed by “ `\\`”, followed by the name of the computer where MainBoss Service is running. After this type a space followed by `query`. For example, if the MainBoss Service computer is called `MBServ`, you would type

```
sc \\MBServ query
```

If this works correctly, you’ll get a list of all the services running on the MainBoss Service computer. This means you can start and stop services on that computer. Otherwise, you’ll get an error message, indicating that conditions for starting and stopping services haven’t been met.

General Troubleshooting

In addition to all the specific troubleshooting points listed in the appendix, there are a number that always apply to Windows programs.

□ Check for viruses. Viruses can cause almost any kind of strange behavior.

□ Check your disks for hardware problems. The most thorough way of doing this is using the `CHKDSK` command. In a command prompt window, type

```
chkdsk /r /f
```

Then reboot your computer. The boot process will do a lengthy but thorough check to find disk errors and recover from them.

□ Check your event logs. You do this starting with the **Control Panel**. Go to **Administrative Tools**, then select **Event Viewer**. Check especially for errors and warnings—these may indicate problems with your computer (e.g. hardware errors or viruses).

If you expand **Window Logs** in the left-hand panel and then click **System**, the center panel will list recent events for your computer. Any marked with red error signs are particularly serious and may indicate critical problems with your hardware or software. Remember, use Google to look up any error messages.

Appendix C: MainBossWeb

This appendix offers instructions on setting up a web site for use with the MainBoss **Web Requests** and **Web Access** modules. These modules provide the ability to deal with work orders and requests through the Internet. The web-based facilities of MainBoss are supported by software called **MainBossWeb**.

> Before setting up MainBossWeb, first make sure that MainBoss is working properly through direct network access (as described in the main body of this guide).

MainBossWeb System Requirements

MainBossWeb is an ASP.NET web application which must run on a Windows system. This means that the web site used by MainBossWeb must be hosted on a Windows system and must use the IIS web server.

We recommend that you use the latest version of IIS. At present, this is IIS 7.5.

Thinkage recommends that you do not install MainBossWeb on the workstation versions of non-server Windows systems. (If you *do* install MainBossWeb in this way, the computer will require at least 500 MB more processor memory.)

Whatever version of IIS and Windows you use, you should use Windows Update to keep the software up to date.

Minimum Server Configuration

It is beyond the scope of this guide to discuss complete configuration of a Windows web server. However, the following list gives a recommended minimum of what is required in the web server. (Note that different versions of Windows use slightly different names for certain configuration options. However, you should still be able to tell which options MainBoss requires.)

- Required server roles (specified by using the **Add Roles** wizard during initial Windows configuration or through the functions of Manage Computer):
    - Web Server (IIS)
    - Application Server
- For the role services of Application Server, select **Web Server (IIS) Support** (and all the required services).
- Under Application Development Features, you need both ASP.NET 3.5 *and* 4.5, plus .NET Extensibility 4.6.

    - Windows 8 gives you the choice of selecting either ASP.NET 3.5 or 4.5; you should select both. (Microsoft Report Viewer requires version 3.5; MainBoss requires 4.6.)
    - Older versions of Windows don't give you a choice; for example, they only specify ASP.NET and .NET Extensibility (without versions). In this case, you automatically get both of the versions you need.

- For the role services of Web Server (IIS), select all the defaults dictated by the previous settings. This includes the following:

```
Common HTTP Features:
      Static Content
      Default Document
      Directory Browsing
      HTTP Errors
      HTTP Redirection
Application Development Features
      ISAPI Extensions
      ISAPI Filters
      .NET Extensibility 4.? (see below)
      ASP.NET 4.? (see below)
Health and Diagnostics
      HTTP Logging
      Logging Tools
      Request Monitor
      Tracing
Security
      Basic Authentication
      Windows Authentication
      Digest Authentication
      Client Certificate Mapping Authentication
      IIS Client Certificate Mapping Authentication
      URL Authorization
      Request Filtering
      IP and Domain Restrictions
Performance
      Static Content Compression
      Dynamic Content compression
Management Tools
      IIS Management Console
      IIS Management Scripts and Tools
      Management Service
```

Note that the `.NET Extensibility` and `ASP.NET` options may or may not be present in the list of possible Windows features—it depends on your Windows configuration. Also, the number associated with these options may be 4.5, 4.6 or 4.7. If the options aren't present, their absence won't affect you; however, if the options are present, they should be turned on.

Also note that with IIS 7.5 running on Windows Server 2008 R2, some of the above are *not* selected by default. In particular, “HTTP Errors” and “HTTP Redirection” are not selected by default. Therefore, you must activate them manually.

License Keys

Before you begin setting up MainBossWeb, record your **Web Requests** and/or **Web Access** license keys in the MainBoss database. To do this, follow the instructions in *Entering License Keys* on page 9.

You cannot use **Web Requests** or **Web Access** without appropriate license keys. To obtain such keys, contact your MainBoss dealer or email `info@mainboss.com`.

Initial Set-Up

Create a folder which will hold all the files required to support the web site. By default, IIS runs from a folder named `C:\inetpub`; therefore, you might run MainBossWeb from a subfolder called

```
C:\inetpub\wwwroot\MainBossWeb
```

You obtain the MainBossWeb software by downloading it from our `mainboss.com` web site. Follow the instructions on the web page to download the necessary files and to copy them to an appropriate folder.

In setting up IIS, follow Microsoft’s documentation. Considerable differences exist between the various Windows systems that might be used. For a discussion of IIS configuration, see

`https://msdn.microsoft.com/en-us/library/ms178477.aspx`

The above web page contains links that explain how to start the IIS manager on supported versions of Windows.

Note: The rest of this appendix applies to set-up and configuration using IIS 7 or 7.5 on Windows Server 2008 (SP2 or R2).

Security Considerations

Before proceeding to configure your MainBossWeb web site, you must make some decisions about security. This section discusses various factors that you should consider.

Depending on your ways of working, especially with cell phones and similar devices, you may find you have to sacrifice some level of security in order to get anything done. You may decide this means you won't use cell phones with MainBossWeb; alternatively, you may decide to live with reduced security or other complications. Thinkage takes no responsibility for any problems you may encounter if you opt for reduced security.

Web Requests: In order to use the **Web Requests** module, users must type in their email address and send it to the MainBossWeb web page. There are two possibilities:

- If you will only accept requests from people in your local Active Directory, then you must set up your MainBoss Service configuration accordingly. In particular, you might use the **Automatically Create Requestors From Active Directory** option.
- If you will accept requests from people outside your Active Directory, then you must enable anonymous authentication as described in *Authentication Settings* on page 105. In this case, anyone who can access the web page is authorized to type in an email address and submit it. Processing only continues if the email address is authorized to submit requests.

  If you do not have the **Web Requests** module, you do not have to allow anonymous authentication to this particular facility.

Web Access: In order to use the **Web Access** module, your MainBossWeb web site must be able to authenticate people as Windows users on the system where the web site runs. This means that people must enter their Windows login names and passwords.

The safest way to do this is to use ASP.NET's integrated Windows Authentication. When users attempt to use the **Web Access** module, they will be asked to send the login names and passwords, which will be securely encrypted when transmitted to the MainBossWeb site.

Unfortunately, integrated authentication does not work with some cell phone and PDA services. Whether or not the device itself can handle encryption, the service provider may use proxy servers that do not support integrated authentication. (This may mean that the device works fine when connected to your own Wi-Fi network but not when connecting through the device's usual service provider.) The symptom of this is that you aren't asked for your login name and password; you simply get a "permissions denied" message.

If the cell phones or PDAs that you intend to use can't handle integrated authentication, you must consider your options.

- One choice would be not to support access through these devices—users would only access MainBossWeb through networked computers or through handheld devices that *do* allow integrated authentication.
- The other choice would be to use "classical" authentication. If you go this route, you have two further options:
  - Using SSL (Secure Socket Layer) or TLS (Transport Layer Security). Users will access the MainBossWeb web site with `https` rather than

`http`, as in

`https://www.ourorg.com/MainBossWeb`

If the device supports this facility, login names and passwords will be encrypted when transmitted.

In order to take this approach, IT personnel will be required to set up SSL/TLS access, which includes the use of a security certificate. See below for more details.

- No encryption. In this case, login names and passwords will be sent in plain text through wireless connections. This leaves you seriously exposed to wireless eavesdroppers, and is therefore not recommended.

Security Certificates: A security certificate may be obtained from a trusted Certification Authority (CA) or may be self-generated. Large organizations often have a CA certificate already, in which case the same certificate may be used for MainBossWeb.

You can create your own self-generated security certificate using the IIS 7 manager. (See *Initial Set-Up* on page 98 of this guide for a reference on how to start the IIS manager.) Once you've started the manager, click the entry for the server in the left-hand panel, then click **Server Certificates** in the **IIS** section of the middle panel. In the resulting window, click **Create Self-Signed Certificate** (in the right-hand panel) to create a self-signed certificate.

If you use a self-signed certificate, devices using `https` to connect to your MainBossWeb web site must be told to trust this certificate. Note that most browsers display strong warning messages when a user first tries to connect with a web site that has a self-signed certificate; therefore, users must be reassured that connecting with your site really is secure. Usually, this is only a problem for users outside your organization, since there are ways to tell all the computers on your internal network to accept a particular certificate.

If you are using SSL/TLS, make sure that your firewall allows such communications through. Typically, SSL/TLS use port 443, so the firewall should allow connections to this port (if you wish to open your web site to outside connections).

Application Pools

Roughly speaking, an application pool consists of one or more web applications that share processing time together as a single process. You must decide whether to put MainBossWeb into an existing application pool or to create a new pool that only contains MainBossWeb. The following considerations apply:

- Each application pool takes up a considerable amount of memory. Therefore, you can reduce memory requirements if you put MainBossWeb into an existing pool.
- Since all applications in a pool must share the processing time allocated to that pool, MainBossWeb will have to share time with everything else in the pool. In particular, if you put MainBossWeb into a pool that contains a high-traffic process, the performance of

MainBossWeb will suffer—the high-traffic process will monopolize allotted processing time.

Perhaps the ideal is to put MainBossWeb into an existing pool whose other members only have a modest amount of traffic.

.NET VERSION: When you create an application pool, you specify which .NET version it uses. All applications within the pool must use the same .NET version. Thus, you must put MainBossWeb into a pool that uses .NET 4.6. Your site may have other ASP.NET applications which use different versions of .NET (e.g. .NET framework v2.0.50727); you cannot put MainBossWeb into such a pool. Furthermore, you should not change the .NET version of any existing pools, since any applications within those pools will stop working.

Application Pool Identity: Each application pool must have an *identity*. This corresponds to a login name that will be used to run all the applications in the application pool. The login name must have sufficient permissions to run all the applications in the pool. In particular, the identity login name for the pool that contains MainBossWeb must have all the permissions required by MainBoss. The name must also have all the SQL Server permissions required to access the MainBoss database.

Important: Whatever application pool you use for the web site, we recommend that it should run in the *Integrated* managed pipeline mode. This ensures appropriate authentication security. However, as discussed in *Security Considerations* on page 98, some cell phone and PDA services do not support integrated authentication, so you may have to choose a different authentication scheme.

To work with application pools, you use the IIS manager. Start the manager on the computer where you will run the web server. (See *Initial Set-Up* on page 98 of this guide for a reference on how to start the IIS manager.)

If you decide to create a new application pool for MainBossWeb, you must create the pool:

1. In the left-hand panel of the IIS manager, expand the entry labeled with the name of the computer.
2. Right-click on **Application Pools**. In the resulting menu, click **Add Application Pool**.
3. In the resulting window, enter a name for the pool (e.g. `MainBoss`). The .NET Framework version should be the highest available and the managed pipeline mode should be `Integrated` (or `Classical` if you make that security choice). When you click **OK**, the pool will be created, using default settings.
4. The new pool is automatically given a default identity. Typically, this is either **NetworkService** or **ApplicationPoolIdentity**. If the default identity is not **NetworkService**, you must change the pool's identity to **NetworkService** by following these steps:
   - Right-click on the name of the pool in the list of application pools.
   - In the resulting menu, click **Advanced Settings**.

- In the resulting window, click the entry for **Identity** (under **Process Model**).
- Click the “...” button on the right-hand side of the field.
- In the resulting window, click **Built-in Account**, drop the arrow, and select **NetworkService** from the list.
- Keep clicking **OK** until you get back to the IIS manager.

Giving Folder Permissions to Network Service or the Application Pool Identity

Whether you choose to run under **NetworkService** or an application pool identity, you must make sure that the identity’s login name has permission to read/write the folder that contains MainBossWeb files. For example, suppose you installed the files in

```
C:\inetpub\wwwroot\MainBossWeb
```

Then you must give the identity permissions on this folder. To do this, follow these steps:

1. Use Windows Explorer to locate the folder that contains the MainBossWeb files.
2. Right-click on the folder name; in the resulting menu, click **Properties**.
3. In the **Security** tab of the resulting window, click the **Edit** button.
4. In the resulting window, click **Add**.
5. In the resulting window, under “Enter the object names to select”, enter `Network Service` (with a space between the words) or else the application pool identity.
6. Click **Check Names** to have Windows verify the specified name.
7. Click **OK** to return to the previous window.
8. With the appropriate name selected, click “Full control” in the list of permissions.

Keep clicking **OK** until you return to Windows Explorer.

Giving SQL Server Permissions to Network Service or the Application Pool Identity

In order for the appropriate identity to access the MainBoss database, it must be given appropriate permissions within SQL Server. To do so, follow these steps:

1. Start SQL Server Management Studio (or Management Studio Express). Specify the instance of SQL Server that holds the MainBoss database.
2. In the left hand column of Management Studio, expand the entry for **Security**.
3. Right-click on **Logins**, then click **New Login**.
4. In the resulting window, click **General** in the left hand column.
5. In the right hand part of the window, click **Search**.

6. In the resulting window, under “Enter the object names to select”, enter `Network Service` (with a space between the words) or else the application pool identity.
7. Click **Check Names** to have Windows verify the specified name.
8. Click **OK** to return to the previous window. You should see an appropriate identity filled into the “Login name” field.
9. In the left hand column of the window, click **User Mapping**.
10. In the right hand part of the window, checkmark the checkbox associated with the MainBoss database.
11. Under “Database role membership”, checkmark the checkbox associated with **MainBoss**. (The “public” checkbox should be checkmarked automatically.)
12. Click **OK** to return to Management Studio, then exit Management Studio.

Web Sites vs. Applications

The next decision to make is whether you want to implement MainBossWeb using a separate web site or an application within an existing web site.

- A separate web site has its own unique IP address and/or port number on the server computer.
- An application uses the same IP address and port number as an existing web site on the server computer. (Loosely speaking, an application is a type of *virtual directory*.)

If you already have a web site on the server computer, we recommend that you implement MainBossWeb as an application under that web site. The only reason to give MainBossWeb its own separate web site is if the server computer doesn’t have any existing web sites, or if there is an incompatibility between the needs of MainBossWeb and the configuration of the existing sites.

Adding and Configuring an Application

This section describes what to do if you decide to implement MainBossWeb as an application under an existing web site.

> As noted in a previous section, you should have copied the MainBossWeb files to a subfolder under the folder used by the IIS web service. If you have not done so, do so now.

Start the IIS manager. In the left-hand panel of the IIS manager, expand the entry for the server computer, then click on **Sites**. Expand the entry for sites until you see the name of the web site under which you will put MainBossWeb.

In order to add MainBossWeb to this web site as an application, **you must stop the web site** (i.e. shut down the web site temporarily). To stop the web site, click on its entry in the left-hand panel of the IIS manager, then click **Stop** in the far right-hand panel.

Once you've stopped the web site, right-click on the web site's entry in the left-hand panel. In the resulting menu, click **Add Application**. This opens a window where you can specify information on MainBossWeb.

- In "**Alias**", specify a name for the application. This is simply a name that will be displayed in the IIS manager to refer to this application; it can be any name that isn't already in use.
- Click the **Select** button associated with the "**Application pool**" field and specify the name of the application pool you have chosen for MainBossWeb—either an existing pool or one you created expressly for MainBossWeb.
- In "**Physical Path**", specify the folder that contains the MainBossWeb files (e.g. `C:\inetpub\wwwroot\MainBossWeb`).
- Click **OK** to close the window.

Once you have configured the web site, you can restart the web site. In the left-hand panel, click the entry for the web site that contains the MainBossWeb application. Then, in the far right-hand panel, click **Start**.

Adding and Configuring a Web Site

This section describes what to do if you decide to implement MainBossWeb as a separate web site.

As noted in a previous section, you should have copied the MainBossWeb files to a subfolder under the folder used by the IIS web service. If you have not done so, do so now.

Start the IIS manager. In the left-hand panel of the IIS manager, expand the entry for the server computer, then click on **Sites**. In the far right-hand panel, click **Add Web Site**. This opens a window where you can specify information on MainBossWeb.

- Click the **Select** button associated with the "**Application pool**" field and specify the name of the application pool you have chosen for MainBossWeb—either an existing pool or one you created expressly for MainBossWeb.
- In "**Physical Path**", specify the folder that contains the MainBossWeb files (e.g. `C:\inetpub\wwwroot\MainBossWeb`).
- In "**Host Name**", enter the name through which users will access your MainBossWeb site.
- In "**Site Name**", enter `MainBossWeb` (or some other suitable name). This name will be used in the URL for accessing the MainBossWeb services. Specifically, users will access MainBossWeb using the URL

http://HOSTNAME/SITENAME

Therefore if “**Host Name**” is `ourcomputer.ourdomain.com` and “**Site Name**” is `MainBossWeb`, users will access the software with the URL

http://ourcomputer.ourdomain.com/MainBossWeb

- In “**Binding**”, specify the appropriate information (e.g. the port number the web site should use). The default port number is 80, and we strongly recommend that you use this default—anything else will make the URL more complicated, since the port number will have to be specified explicitly. (Also, some cell phone services do not support connections to non-standard port numbers.)

  Note that if you’re using SSL or TLS security (as discussed in *Security Considerations* on page 98), you specify `https` as the binding type and then specify an SSL certificate using the drop-down list.
- Click **OK** to close the window.

Once you have configured the web site, you can start it. In the left-hand panel, click the entry for the web site. Then, in the far right-hand panel, click **Start**.

Authentication Settings

Whether MainBossWeb is a separate web site or an application within another web site, you must set its authentication settings (i.e. how the software determines who is or isn’t an authorized user).

In the left-hand panel of the IIS manager, expand **Sites** and click on the name of the MainBossWeb web site or application.

In the middle panel of the IIS manager, the IIS section has an entry for IIS *Authentication*. Double-click this entry to open a window where you can specify authentication settings. We recommend the following:

- If you intend to accept requests from people outside your local Active Directory, enable Anonymous Authentication
- Disable ASP.NET Impersonation (this is the default)
- Disable the Forms Authentication setting
- Enable Windows Authentication

Application Settings

Whether MainBossWeb is a separate web site or an application within another web site, you must set its application settings. These specify which database the software should access.

In the left-hand panel of the IIS manager, click on the name of the MainBossWeb site. In the middle panel of the IIS manager, the ASP.NET section has an entry for *Application Settings*. Double-click this entry to open a window where you can specify the settings.

You will see there are settings for `dbserver` and `dbname`. Initially, these have stand-in names that should be replaced with the real names used by your site.

Click on the entry for `dbserver`. In the far right-hand panel, click **Edit**. In the resulting window, set "**Value**" to the name of the SQL Server that maintains the MainBoss database. This name will have one of the following two forms:

`COMPUTER\INSTANCE`

> The name of the computer where the SQL Server resides and the name of the particular SQL Server instance that maintains the MainBoss database.

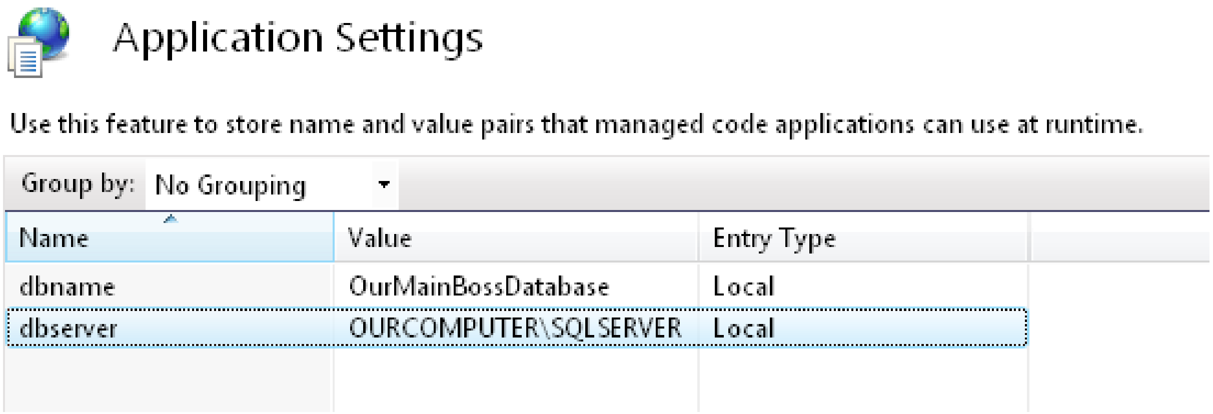
`COMPUTER`

> Just the name of the computer where the SQL Server resides. This indicates that the MainBoss database is maintained by an unnamed instance of SQL Server.

Click **OK** once you have entered the appropriate "**Value**".

Now click on the entry for `dbname`. In the far right-hand panel of IIS manager, click **Edit**. In the resulting window, set "**Value**" to the name of the MainBoss database as maintained by the server specified with the `dbserver` setting. (Note that this is the database name, not the organization name that might be specified by a particular user.) Click **OK** once you have entered the appropriate "**Value**".

When you're done, the application settings window should look something like this:



Application Settings

Use this feature to store name and value pairs that managed code applications can use at runtime.

Group by: No Grouping

Name	Value	Entry Type
dbname	OurMainBossDatabase	Local
dbserver	OURCOMPUTER\SQLSERVER	Local

Important: After setting this information, you may have to stop and start the application pool in order for the settings to take effect. In IIS manager, right-click on the entry for the application pool, then click **Stop**. Wait a moment, then right-click and click on **Start**.

Software Initialization

The first time someone connects with the MainBossWeb web site, the software must compile itself and perform various set-up processes. This can take a long time and use a lot of computer memory. Therefore, you should make the first connection at a time when the increased processor and memory load will not affect important work on the computer.

Once the software has set itself up, it doesn't have to go through the process again unless something significant changes in your configuration, e.g. you install a new version of the web server, SQL Server, or MainBoss.

Accessing SQL Server on a Different Computer

As noted previously, configuring MainBossWeb is easier if the web server is on the same computer as the SQL Server that manages MainBoss. If the two are on different computers, a few considerations apply:

- MainBossWeb works with MainBoss Service. As discussed in *SQL Server and MainBoss Service* on page 76, MainBoss Service will run under the `Network Service` login name on the computer where it is installed. To access SQL Server, `Network Service` identifies itself with a login name of the form `COMPUTER$` where `COMPUTER` is the name of the computer. Installing MainBoss Service will therefore put `COMPUTER$` into the login list of the instance of SQL Server that manages your MainBoss database.
- If your web server typically runs web sites in a similar manner (running under `Network Service`), then you should create a similar `COMPUTER$`-style login name for the computer in SQL Server's login list. Assign the same SQL Server security roles to this name as for the name associated with MainBoss Service. (If MainBossWeb runs on the same computer as MainBoss Service, you don't have to do anything—the appropriate login name will already be registered with SQL Server.)
- For recent releases of IIS Server, the default behavior is not to run web sites under `Network Service`. Instead, the application pools containing web sites run under special domain login names created specifically for the application pool. If you intend to run your MainBossWeb web site in such an application pool, you must manually add the special domain login name to the list of login names recognized by SQL Server. Do this with SQL Server Management Studio. Give the login name the same security roles as the login name associated with MainBoss Service.

Integrating MainBossWeb with MainBoss Service

If you are using the MainBoss Service module, you should adjust its configuration so that the service knows about MainBossWeb. To do this, follow these steps:

1. Login under a user name with Windows Administrator and MainBoss Administration permissions.
2. Start MainBoss.
3. Go to **Administration** | **MainBoss Service** | **Configuration**.
4. Double-click the single record in the table viewer.
5. In the resulting window, go to the **Outgoing Mail** section.
6. Set the field "**MainBoss Web Access URL**" to the URL for your MainBossWeb web site, e.g. `http://ourcompany.com/MainBossWeb`.
7. Click **Save and Close**.
8. Go to **Administration** | **MainBoss Service**.
9. Click **Configure the Windows Service for MainBoss**. This updates MainBoss Service to use your new configuration.
10. Click **Restart Service** to stop the service and start it again with the new configuration.

If you do not do the above, messages sent out by MainBoss Service may contain URLs that do not work.

Unit Location Autocomplete

Your organization has the option of using autocomplete for the "**Unit Location**" field in a request submitted via the web. The user starts by typing a few characters; the web page then downloads a list of units that contain those characters and the user can pick the appropriate unit from the list.

Autocomplete makes it easier for users to specify a unit. However, there are some potential disadvantages. If you have a large number of units whose names contain the characters that the user typed in, the user will be presented with a long list of names; this can be difficult for users to use, especially if they are submitting their requests using smart-phones with relatively small screens. It is also possible that the characters typed by the user don't match any known units; the user may have to make several attempts before guessing an appropriate name.

If your organization chooses not to use autocomplete, the "**Unit**" field in the created request will contain whatever the user typed in, whether or not it matches any known unit name.

If you wish to change your current autocomplete setting, you must edit a file associated with your web site. For details, see https://www.mainboss.com/english/support/faq/mb30_autocomplete.shtml.

MainBossWeb Troubleshooting

In our experience, the most common problem that users encounter is not having Microsoft's .NET 4.6 installed on the computer where the web site is running. This leads to numerous errors. .NET 4.6 is available free from Microsoft—use Google or some other search engine to search for ".NET 4.6 download", then download the software from Microsoft and install it on the web server computer.

This rest of this section lists a number of error messages you might see when trying to use MainBossWeb, plus ways to correct the problems.

```
There was a problem validating access to the database XXX on server
           YYY. The database-principal 'dbo' does not exist or user
           is not a member. Failed SQL command is: declare @result
           int; exec @result = sp_getapplock 'MainBossSession',
           'shared', 'session', 0, 'dbo'; select @result
```

This indicates you did not give permissions to the login name for the identity of the application pool where MainBossWeb runs. Follow the instructions in *Application Pools* on page 100.

```
An error occurred while processing your request. Database server
           'ServerMainBossDatabaseIsOn' was not found or was not
           accessible.
```

This indicates that you did not change the default `dbserver` application setting. See *Application Settings* on page 105 of this guide.

```
Could not load file or assembly 'System.Web.Mvc' or one of its
           dependencies. Access is denied.
```

This indicates that you did not give `DOMAIN\IIS_MAINBOSS` read/write file system privileges to the directory containing temporary ASP.NET files. See the previous section of this guide for more information.

Note: Whenever you encounter diagnostic messages, you can usually find useful information by typing them into Google and searching for information.

The Windows Firewall: In our experience, many of the difficulties that arise when configuring MainBossWeb are due to firewall problems. If the Windows firewall is not adjusted to allow various information to pass through, MainBossWeb will not work. Firewall problems are especially difficult to diagnose because they generate no error messages or log entries; you just don't see anything happening.

Requestor Rejection Errors: Requests may be rejected for a variety of reasons. For example, suppose you have reached the limit of authorized requestors allowed in the **Requestors** table according to your license keys. If a new requestor attempts to submit a request, the request will be rejected, even though the person would otherwise be accepted according to your MainBoss Service configuration options.

Cookie Usage

Cookies used by MainBossWeb are stored as subkeys under the cookie key `MainBossWeb`. The following subkeys are defined:

`requestorEmail`
: The email address that a user last entered, provided that the address was recognized as a valid requestor in MainBoss.

INDEX