Table of Contents

1 Document Overview

Author: NETIKUS.NET ltd
Date: 29th Sep 2006
Revision: 1.10

Automated Reliable Windows Backup using EventSentry and NTBackup

Title	Reliable Backup using EventSentry and NTBackup
Summary	How to configure NTBackup to perform a backup and receive an email when the backup was not run successfully. Optionally log backup activity to a database.
Benefits	Using EventSentry (pricing starts at USD 69.00) to monitor NTBackup is more cost effective than purchasing 3rd party backup software which usually costs several hundred dollars (provided that NTBackup does provide all the features you need).
Software	MS Windows 2000, MS Windows Server 2003, Windows XP EventSentry v2.72 or later
Hardware	Not applicable
Skill Level	Beginner - Intermediate
Skills Required	- Basic understanding of Windows NT, 2000, 2003 or XP - Basic understanding of batch files
Download	http://www.netikus.net/ (guides section)

2 Document Details

Overview	This document describes how to setup an automatic backup using NTBackup that will automatically email you when the backup did not complete successfully. You can optionally also log all backup activity to a database and query the database through the EventSentry web reports.
Additional Benefits	You can also configure EventSentry to notify you when similar recurring update tasks (e.g. virus definition updates) did not occur during specified time intervals, plus monitor the event logs and system health of the server. This guide will also give you a good insight into how EventSentry works and the many different scenarios its flexible filter system can be applied towards.

EventSentry	EventSentry is an event log, system and network monitoring suite that is available both as a commercial edition ("EventSentry") and a freeware edition ("EventSentry Light"). You will need either the full or trial edition of EventSentry for this guide, the light edition is not sufficient as it does not support recurring events.
Why?	This guide was written to help current and future EventSentry customers setup a reliable yet affordable backup system.

3 EventSentry Setup

3.1 Download & Installation

Downloading EventSentry

You can skip this chapter if you have already downloaded EventSentry. If you have already purchased EventSentry then you can download the latest version from http://www.eventsentry.com/downloads_downloadnow.php.

You can download a full evaluation copy of EventSentry (will run for 45 days, extensions available) from http://www.eventsentry.com/downloads_downloadtrial.php.

You will not be able to use EventSentry Light for the functionality described in this guide, as EventSentry Light does not support any database-related features.

Installing EventSentry

After downloading the latest version, simply run the setup (**eventsentry_setup.exe**). During setup make sure that you enter the correct email (SMTP) information and select at least the "Event Log Agent" and the "Management Application". All other features are optional and can be added at a later time by running the setup again.

If you would like to record backup activity to a database then you need to make sure that you select "Database Features" and the appropriate sub feature (when using MS Access or MS SQL Server).

Testing EventSentry

After setup has completed EventSentry should automatically be configured to email you all errors, warnings and audit failures via email. To make sure that EventSentry and the email notification (aka as "target") are setup correctly, open the EventSentry management application and click on "Test Agent" in the "Service Control" container. You should receive an email with two test entries within the next minute.

Please refer to the EventSentry manual or the EventSentry knowledge base if you are experiencing difficulties setting up EventSentry.

4 Configuring NTBackup

After downloading and installing EventSentry we need to configure NTBackup to perform an automated (e.g. daily) backup. Once the backup is configured and working properly we can configure EventSentry.

Setting up an automated backup involves the following steps:

1. Determining the backup device and backup destination
2. Creating a backup selection file (.bks) using the NTBackup application
3. Creating a batch script
4. Scheduling the backup

4.1 Determine Backup Device & Destination

Determining the name of your backup device

Open up the device manager (right-click "My Computer" -> Properties -> Hardware Tab -> "Device Manager") and expand the "Tape Drives" node. Write down (copy & paste) the names of all backup devices under the "Tape Drives" node (e.g. "Compaq DDS3 12/24 GB DAT Drive"). The name(s) will

later be needed for rsm.exe.

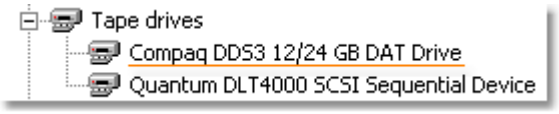


Figure 1

Determining the backup destination
Open the backup application (Start -> Programs -> Accessories -> System Tools -> Backup) and click on the backup tab. Optionally dismiss the wizard as we will not be using the wizard in our example.

Click on the **Backup** tab and review the available backup destinations on the bottom left of the dialog. You will need to specify the appropriate backup destination in the next chapter when creating the batch file.

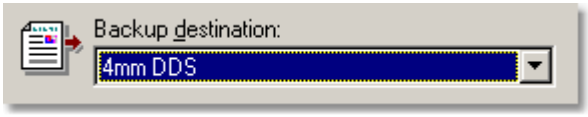


Figure 2

4.2 Creating a NTBackup .bks file

Creating a backup selection
Open the backup application (Start -> Programs -> Accessories -> System Tools -> Backup) and click on the backup tab. Dismiss the wizard as we will not be using the wizard in our example.

Then select all the drives and/or folders that you would like to backup, possibly including System State and/or Exchange Server. When you are done, save your selection by selecting "Save Selections As ..." from the "Job" menu. I recommend that you give your selection a descriptive name such as "FullBackup" or similar.

4.3 Creating a batch script

The next step is to create the batch file that we will later schedule to run the attended backup. You can basically take the template listed in the next chapters and make minor modifications to make it work on your server. Please note that we have two templates, one for a regular tape backup and one for a file backup (when backing up to a file).

Before you create the batch file you will need to determine the following variables. You can click on the underlined names below for more information on how to determine the name.

Type	Variable Name in script file	Example
Backup Device Name (tape backup only)	[BACKUP_DEVICE]	Compaq DDS3 12/24 GB DAT Drive
Backup Selection (bks) File	[BACKUP_FILE]	C:\NTBackup\DailyBackup.bks
Backup Destination (tape backup)	[BACKUP_DESTINATION]	4mm DDS
Backup Destination (file backup)	[BACKUP_DESTINATION]	D:\Windows Backup\

Once you have determined all the required names, replace the strings (e.g. **[BACKUP_DESTINATION]**) in the templates with the actual value (e.g. **4mm DDS**) and save the file in a .cmd file.

Example
For example, let's say you want to backup the selection you saved in **C:\Batch\NTBackup_Daily.bks** file to the **Quantum DLT4000 SCSI Sequential Device**, which is a **DLT** drive. The resulting script

would look like this:

```
REM DOS script to automate backup using ntbackup.exe to a tape device

rsm.exe refresh /LF"Quantum DLT4000 SCSI Sequential Device"

sleep 30

REM Create Backup Name based on date and time
for /f "Tokens=1-4 Delims=/ " %%i in ('date /t') do  set DT=%%i-%%j-%%k-%%l
for /f "Tokens=1" %%i in ('time /t') do set TM=-%%i

set TM=%TM::=-%
set DTT=%DT%%TM%

REM Execute Backup
%SYSTEMROOT%\system32\NTBACKUP.EXE backup "@C:\Batch\NTBackup_Daily.bks" /n "%DTT%"
/d "%DTT%" /v:no /r:no /rs:no /hc:on /m normal /j "%DTT%" /l:f /p "DLT" /um

REM Eject Tape
rsm.exe eject /PF"%DTT% - 1" /astart

exit
```

Save the contents of the file in a directory of your choice, for example **C:\Batch\NTBackup_Daily.cmd**. We schedule this backup to run daily on weekdays in the next chapter.

4.3.1 Template for Tape Backups

```
REM DOS script to automate backup using ntbackup.exe to a tape device

rsm.exe refresh /LF"[BACKUP_DEVICE]"

sleep 30

REM Create Backup Name based on date and time
for /f "Tokens=1-4 Delims=/ " %%i in ('date /t') do  set DT=%%i-%%j-%%k-%%l
for /f "Tokens=1" %%i in ('time /t') do set TM=-%%i

set TM=%TM::=-%
set DTT=%DT%%TM%

REM Execute Backup
%SYSTEMROOT%\system32\NTBACKUP.EXE backup "@[BACKUP_FILE]" /n "%DTT%" /d "%DTT%"
/v:no /r:no /rs:no /hc:on /m normal /j "%DTT%" /l:f /p "[BACKUP_DESTINATION]" /um

REM Eject Tape
rsm.exe eject /PF"%DTT% - 1" /astart

exit
```

4.3.2 Template for File Backups

```
REM DOS script to automate backup using ntbackup.exe to a file

REM Create filename based on date
for /f "Tokens=1-4 Delims=/ " %%i in ('date /t') do  set DT=%%i-%%j-%%k-%%l
for /f "Tokens=1" %%i in ('time /t') do set TM=-%%i

set TM=%TM::=-%
set DTT=%DT%%TM%
set FILENAME=[BACKUP_DESTINATION]%DTT%.bkf
```

```
REM Execute Backup
%SYSTEMROOT%\system32\NTBACKUP.EXE backup "@[BACKUP_FILE]" /n "%DTT%" /d "%DTT%"
/v:no /r:no /rs:no /hc:on /m normal /j "%DTT%" /l:f /f "%FILENAME%" /um
```

4.4 Scheduling the backup

We will use **Scheduled Tasks** (introduced with Windows 2000) to schedule the backup to run daily on weekdays. It is not recommend to use the "at" command to schedule backups since you cannot run the backup under a different user account.

Open the scheduled tasks by navigating to Start -> Settings -> Control Panel -> Scheduled Tasks. Double-click "Add Scheduled Task" and click on the **Next** button which will show the screen below:

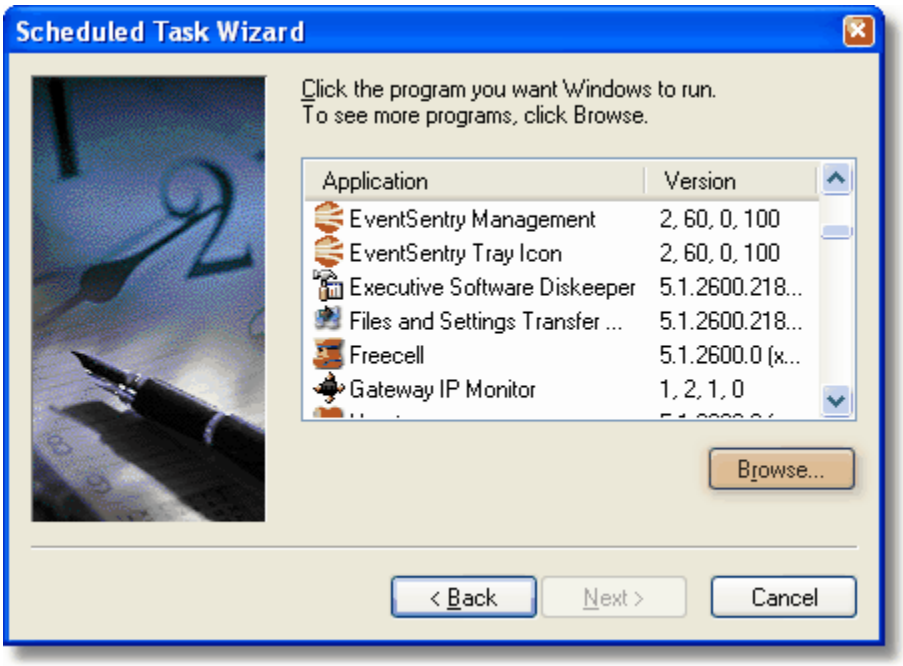


Figure 3

Then, click the **Browse** button and browse to the .cmd file created in the previous chapter. On the next dialog

Figure 4

specify a name for the scheduled task (e.g. **Daily Backup**) and set a basic schedule (we will modify this later). Click the **Next** button and set the time when the backup should be started every day:

Figure 5

Then hit the **Next** button and specify the user account the scheduled task should run under. It is **not** recommended that you use your login here, instead create a user account that is just to be used for backups. This account should have adequate permissions to both access the files that are to be backed up and to access the tape/file device.

Figure 6

On the last dialog click the **Open advanced properties ...** checkbox and click **Finish**. When you are presented with a new dialog click the **Schedule** tab to correct or adjust the schedule.

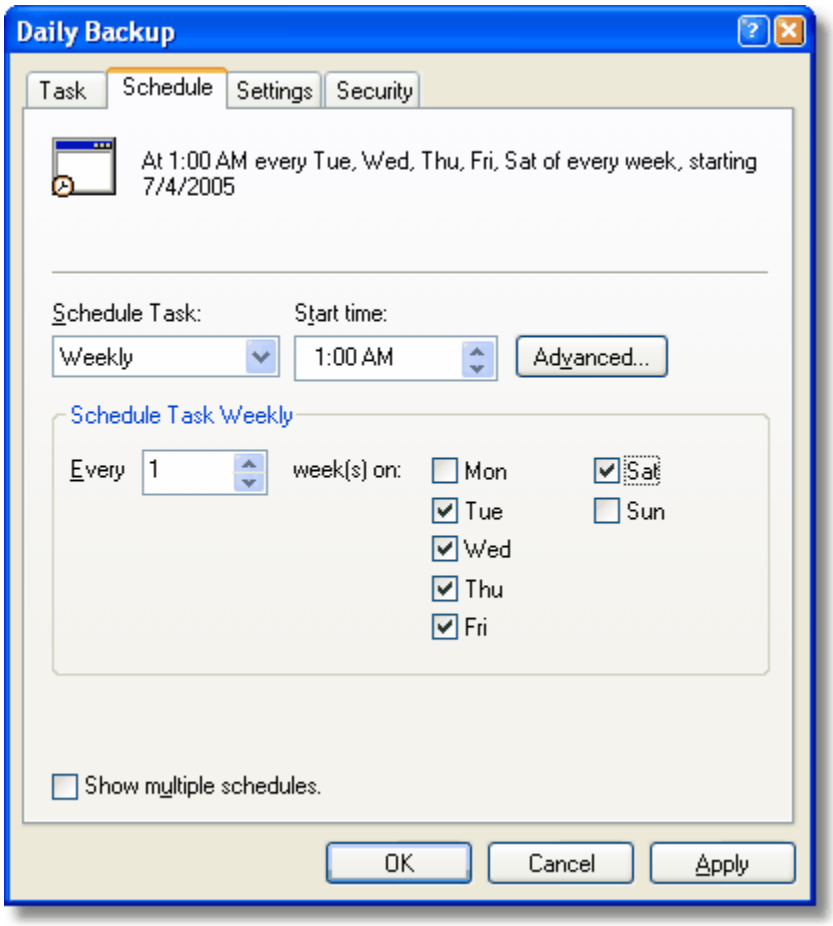


Figure 7

The scheduled task wizard will automatically configure the scheduled task to run from Monday through Friday, however since our scheduled task starts to run after midnight we need to **reconfigure the schedule to start on Tuesday** and run until Saturday.

If you need to change the permissions of this scheduled task object then you can adjust those by clicking the **Security** tab.

And that's it, at this point you should have a scheduled task that runs your backup according to your schedule. If you are backing up to a tape device then the only thing you should need to do now is change the backup tape on a daily basis.

The next chapters will show you how to configure EventSentry so that you will get an email (or a different supported notification such as SNMP) when the backup does not complete successfully in a specified time interval.

5 Configuring EventSentry

You can configure EventSentry to perform two useful things in regards to a system backup:

1. Notify you when the backup does not complete in the specified time interval
2. Log backup activity to a database or text file
3. Receive a weekly summary on Monday morning confirming that the backup ran succesfully

You could configure EventSentry to email you when the backup did complete, but this would mean that you would get another email every day. The first option is more elegant and will notify you immediately when there was a backup problem.

All following instructions will apply to a fresh install of EventSentry 2.60 (or later), but will also work for already customized installations.

5.1 Failed Backup Notification

In our example we will backup the following:

1. The local **C** drive
2. The network share **\\FILESERVER\DOCUMENTS**
3. The information store on server **EXCH-SRV1**

To accomplish our goal we will need to setup the following in EventSentry:

1. **Notification Target (e.g. Email):** You will be notified through this type of notification if the backup didn't work.
2. **Filter Package:** We will place all filters into its own separate filter package.
2. **Threshold Filter:** The threshold filter is necessary to determine that all three backup items (see above) ran.
3. **Recurring Event Filter:** The recurring event filter will log an error if the threshold filter *does not* determine that the backup ran successfully.
4. **Notification Filter:** Usually already setup, will forward the previously logged error via email.

5.1.1 Adding a notification target

Before we start configuring the filter rules we should setup the notification that is to be triggered when the backup did not complete as expected. For our example we will use an email (SMTP) notification.

Open the EventSentry management application and review the installed notifications:

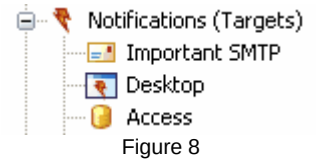


Figure 8

If the desired notification type is already listed there then you can skip this chapter. If it is not, then right-click the **Notifications (Targets)** container and select **Add** from the menu to add a new notification. Give the new notification a descriptive name (e.g. **Email Notification**) and hit ENTER. Then select the **SMTP** tab and configure all required options for the SMTP target, similar to shown below:

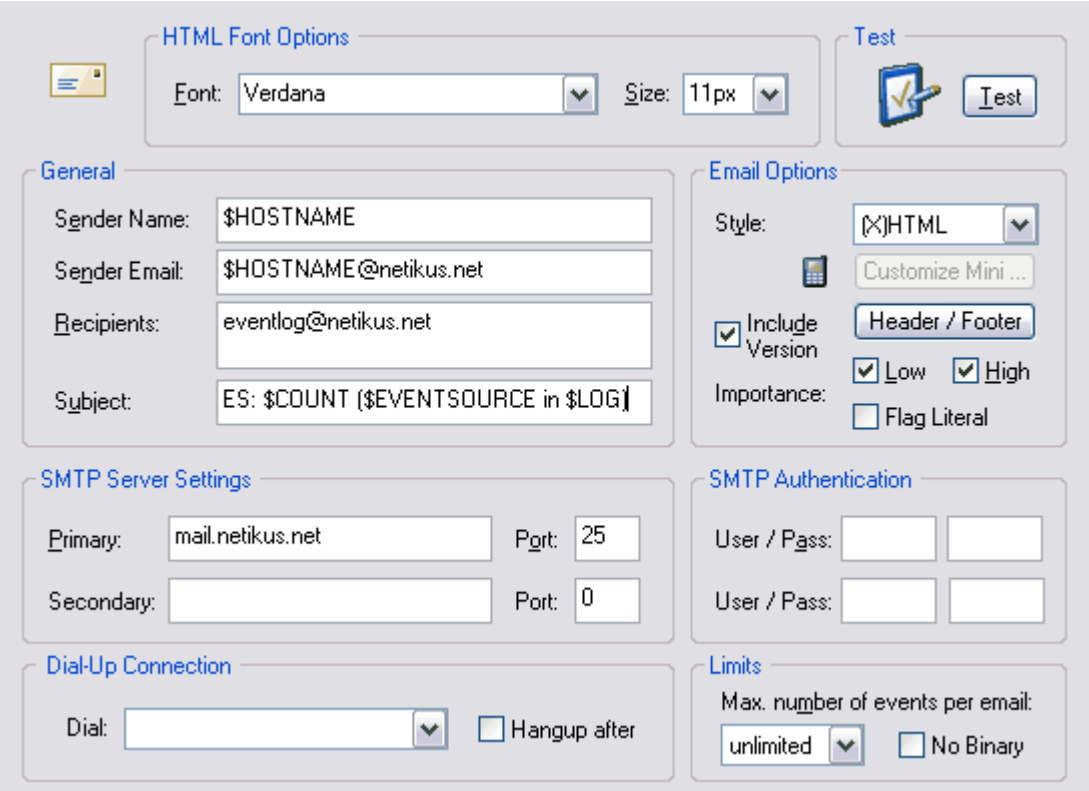


Figure 9

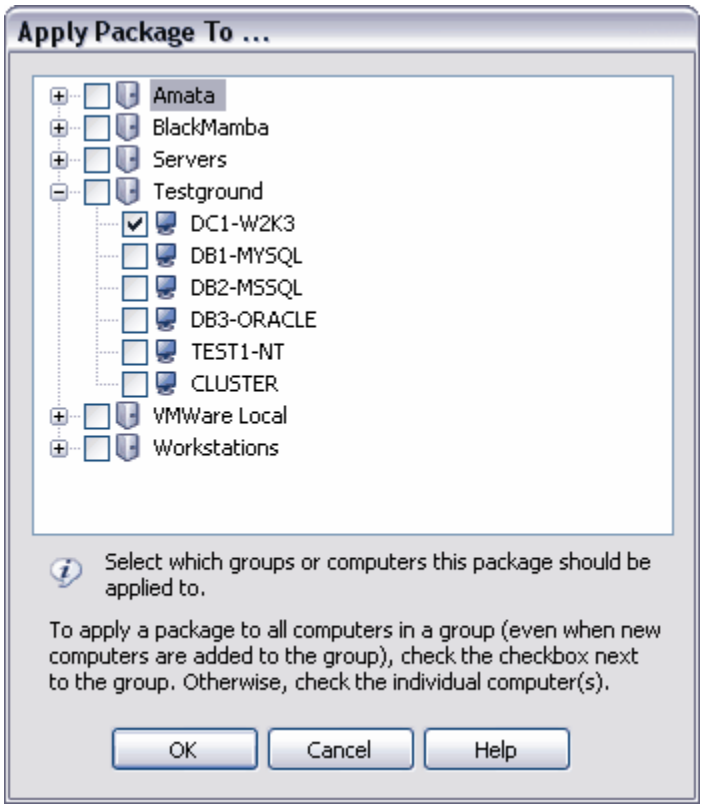
We are now ready to proceed to create the first filter.

5.1.2 Adding a Filter Package

A filter package is used to group one or more filters (usually more than one) into one logical entity that can then be assigned to one or more computers or groups. Since we will be creating multiple filters to monitor our **NTBackup** jobs, it makes sense to group them together, so we will create a new filter package for that purpose.

Creating the Package
In the management console, right-click the **Filter Packages** container and select **Add Package**. When prompted for the name, specify a name like "**Backup**" or similar that describes the function of the filters well.

Assigning the Package
In order for a package to work correctly, it needs to be assigned to at least one computer. As such, right-click the package you just created and select **Assign**. In the resulting dialog, select the computer and/or groups where you will be backing up data. If you only have EventSentry installed on one host, then you can also make the package global (right-click package and select "Global").



5.1.3 Creating a threshold filter

The purpose of the threshold filter is to do one thing: Write an event to the event log when a configurable number of events occur during a specified time period.

NTBackup will log two events to the event log for each logical/network drive (e.g. C:) and logical entity (e.g. System State) it backs up. One event indicating that the backup started, and one event indicating that the backup ended (if it did, that is).

Action	Event Log	Event Type	Event Source	Event ID	Message Text
Backup Start	Application	Information	NTBackup	8000	Begin Backup of '\\SERVER\SHARE' Verify: Off Mode: Append Type: Normal
Backup End	Application	Information	NTBackup	8001	End Backup of '\\SERVER\SHARE' Verify: Off Mode: Append Type: Normal

In our scenario we are only interested in knowing that the backup of each "entity" was successful, as such the 8001 event. Since we are backing up three different entities we are looking to find three events that match the above criteria of event 8001.

In the management application, expand the **Filter Packages** node, right-click the **Backup** filter package and select **Add Filter**. Assign the filter a descriptive name, such as "Daily Backup Threshold" and hit ENTER.

Configure the general filter settings as shown in the screenshot below:

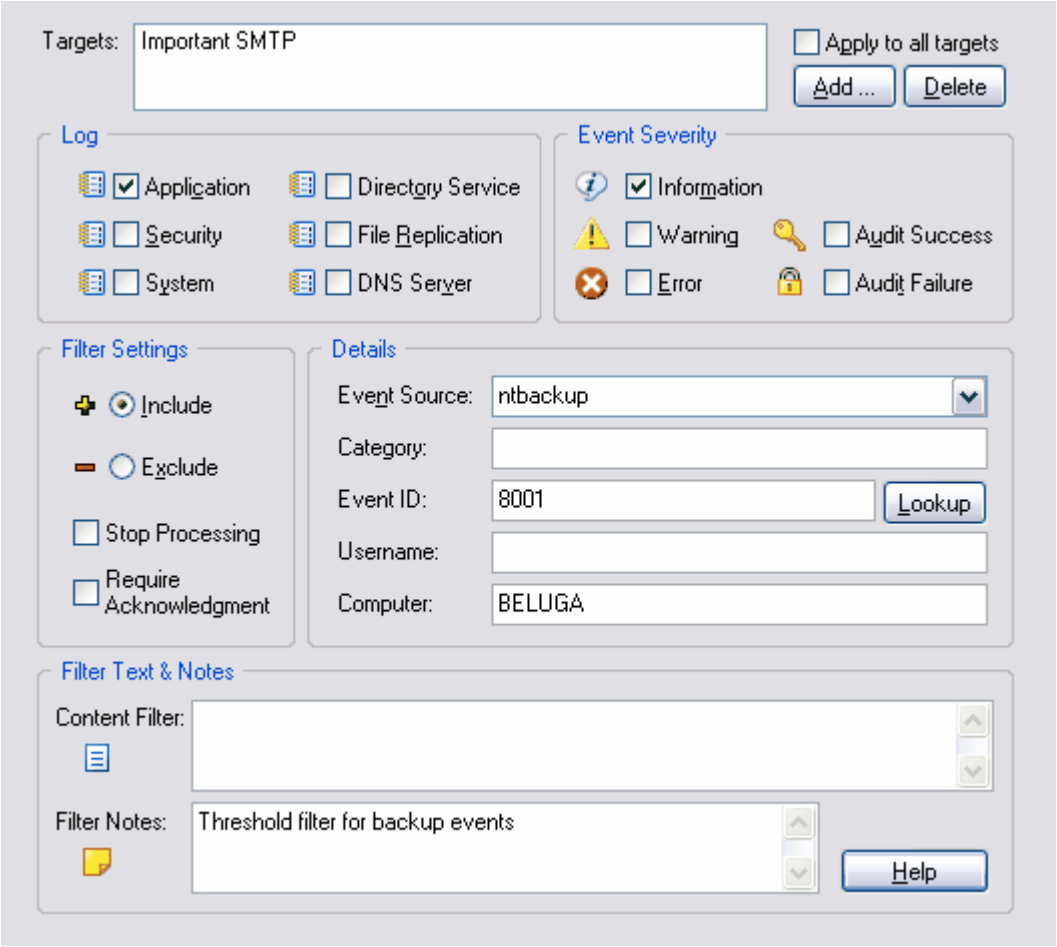


Figure 10

When you are done, click the **Threshold** tab to configure the threshold options. In our scenario we are assuming that our backup will take no longer than 6 hours, but you might have to adjust that period if your backup takes longer or significantly shorter:

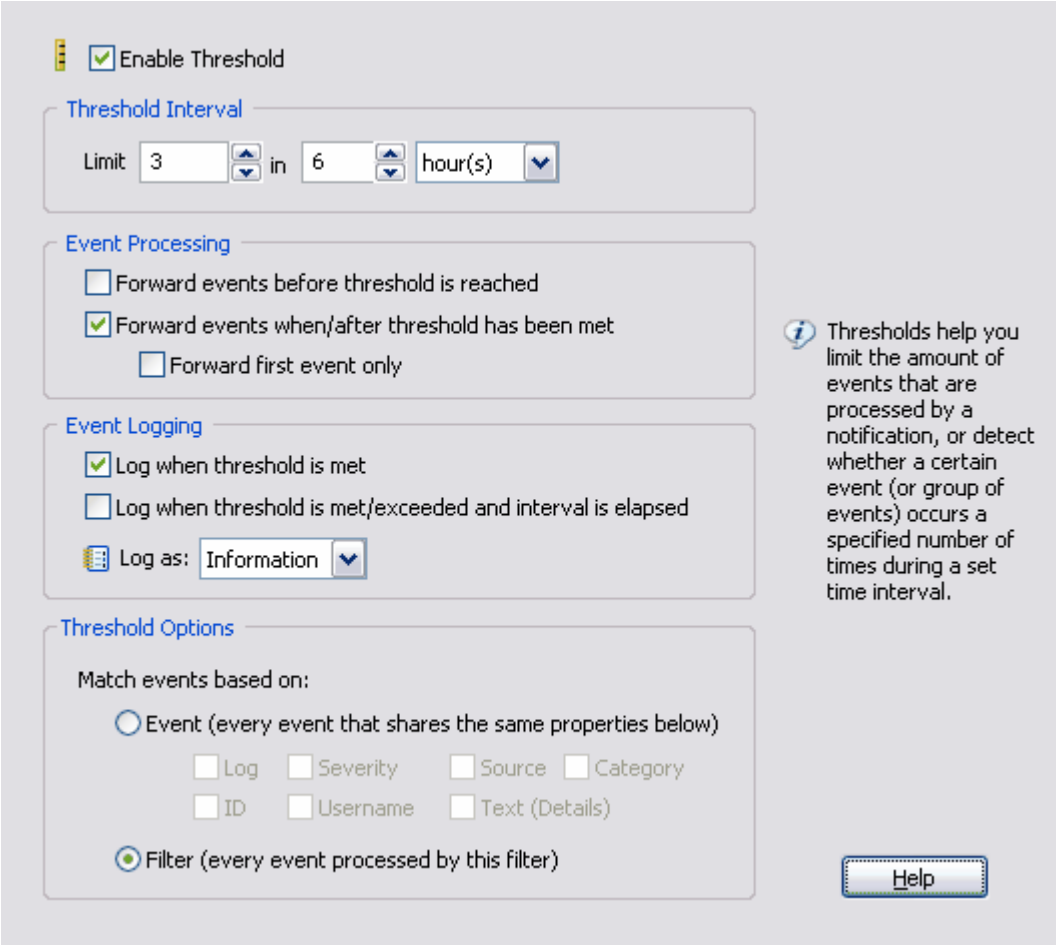


Figure 11

So what are these threshold settings doing? If this filter finds at least 3 events that match the criteria specified above, then it will write the following event to the event log:

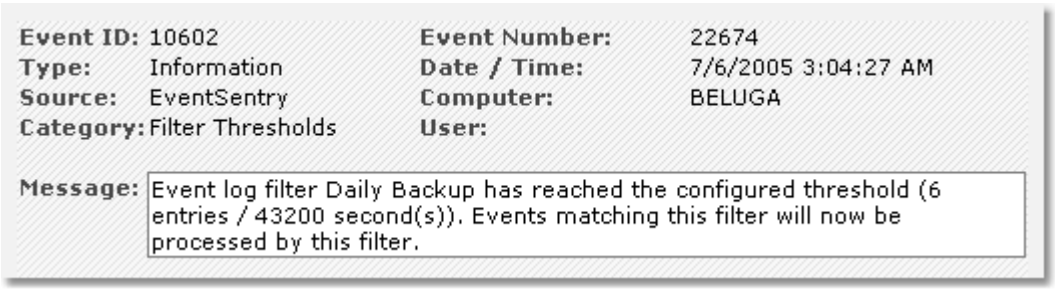


Figure 12

Should our backup produce more than three events in 6 hours, then the events themselves will be forwarded to the email notification specified in figure 10. This is useful when the backup is changed (e.g. an additional drive is monitored) since it can act as a reminder that you need to change the threshold filter and increase the count. *Please note that the threshold filter that triggered the event in figure 12 was configured for 6 entries in 12 hours, not 3 entries in 6 hours.*

So, ideally this threshold filter will never actually forward events to a target (unlike regular filters), but *only count* events based on our settings.

Our last step will be to create a recurring event filter that will log an error to the event log if our threshold event (10602) is not written to the event log, indicating that not enough "backup successfully ended" events were written to the event log.

5.1.4 Creating a recurring event filter

So far we have accomplished the following:

- NTBackup runs daily at the configured times
- An EventSentry threshold filter will write an event to the event log if all backup jobs ran successfully

The last step is to create a recurring event filter that will write an error event to the application event log when the backup did not run or only ran partially.

Right-click the previously created threshold filter in the management application and select **Add Filter**. Enter a descriptive name for the filter (e.g. **Backup OK**) and configure the general filter settings as follows:

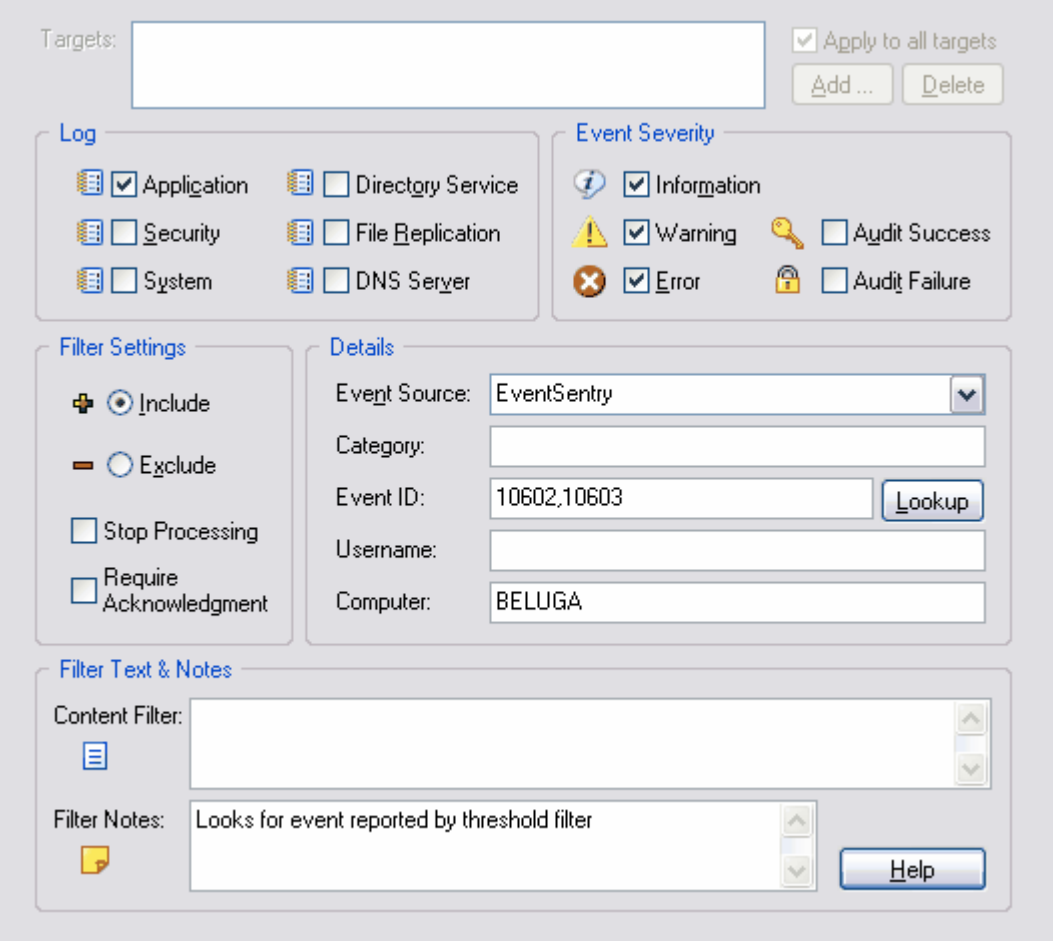


Figure 13

Please note that the targets area will be grayed out after you set the **Schedule Type** to **Recurring Event**. A notification is not needed since a recurring event filter never actually forwards events to a target, it only ensures that an event occurred at a specified time.

Then click the **Day / Time** tab to configure the time period when the backup needs to have been

completed. For example, if the backup starts at 1AM in the morning and typically runs about 3 hours, then we would configure the recurring event filter to look for the threshold event between 3AM and 6AM. This will make sure that we have room for changes (if the backup is faster or slower over time) and still ensure that we are notified by 6AM if the backup did not complete on time.

Please note that the recurring event filter will not log an error until the end of the configured period is reached, provided the expected event did not occur.

You can now set the time period when the backup is expected to be completed by pushing the square buttons as shown below:

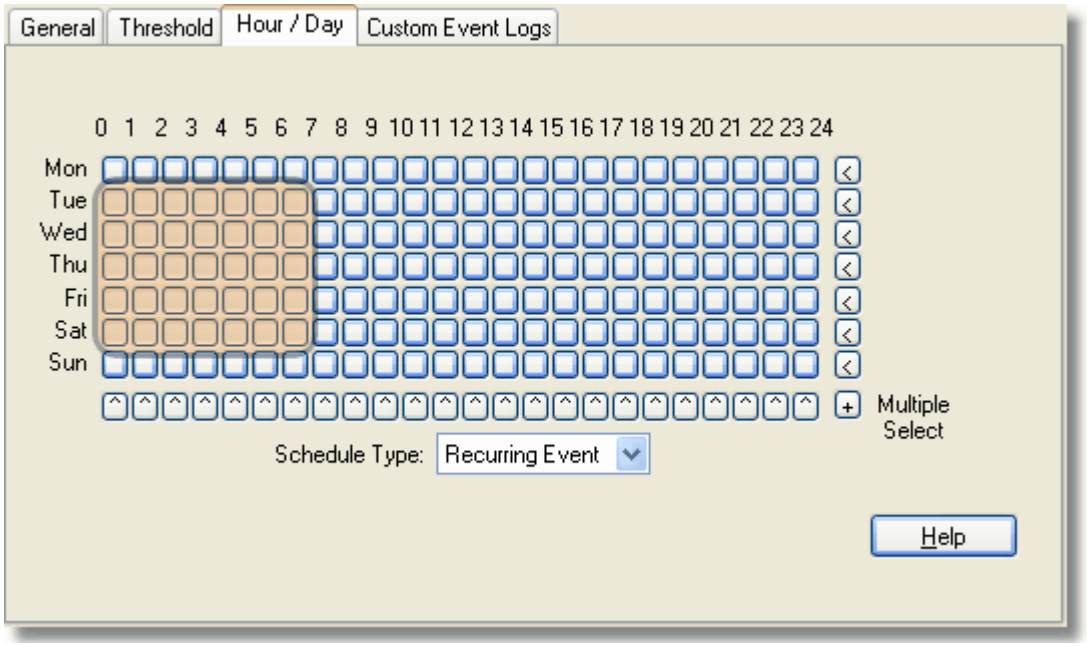


Figure 14

After setting the time period you will need to set the schedule type to **Recurring Event**. So in the example above, EventSentry will expect an event with event id 10602 to be logged to the event log every day from Tuesday to Saturday between midnight and 7AM in the morning. If the backup did not succeed, then an event will be logged at 7AM.

So what happens if the backup didn't run to our satisfaction? You will see the following event in the application event log:

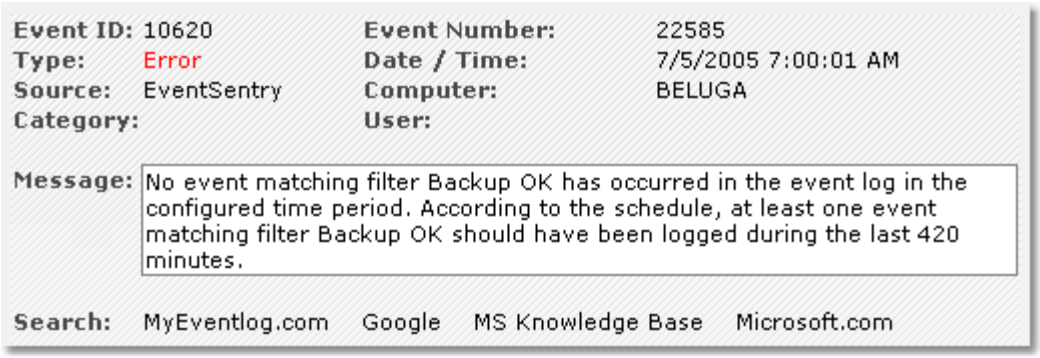


Figure 15

5.1.5 Creating a notification filter

The default installation of EventSentry already contains a filter which forwards all **errors** and **warnings** to an email target. If you do not have a filter that will forward error events from the event logs to you, or if you need an additional notification then follow these steps to create a notification filter. This filter is necessary in order to forward a potential error event (as reported by the recurring event filter) to you via email.

Right-click the previously created filter and select **Add Filter**. Give this filter a descriptive name such as **Forward Errors** and configure the general settings as follows:

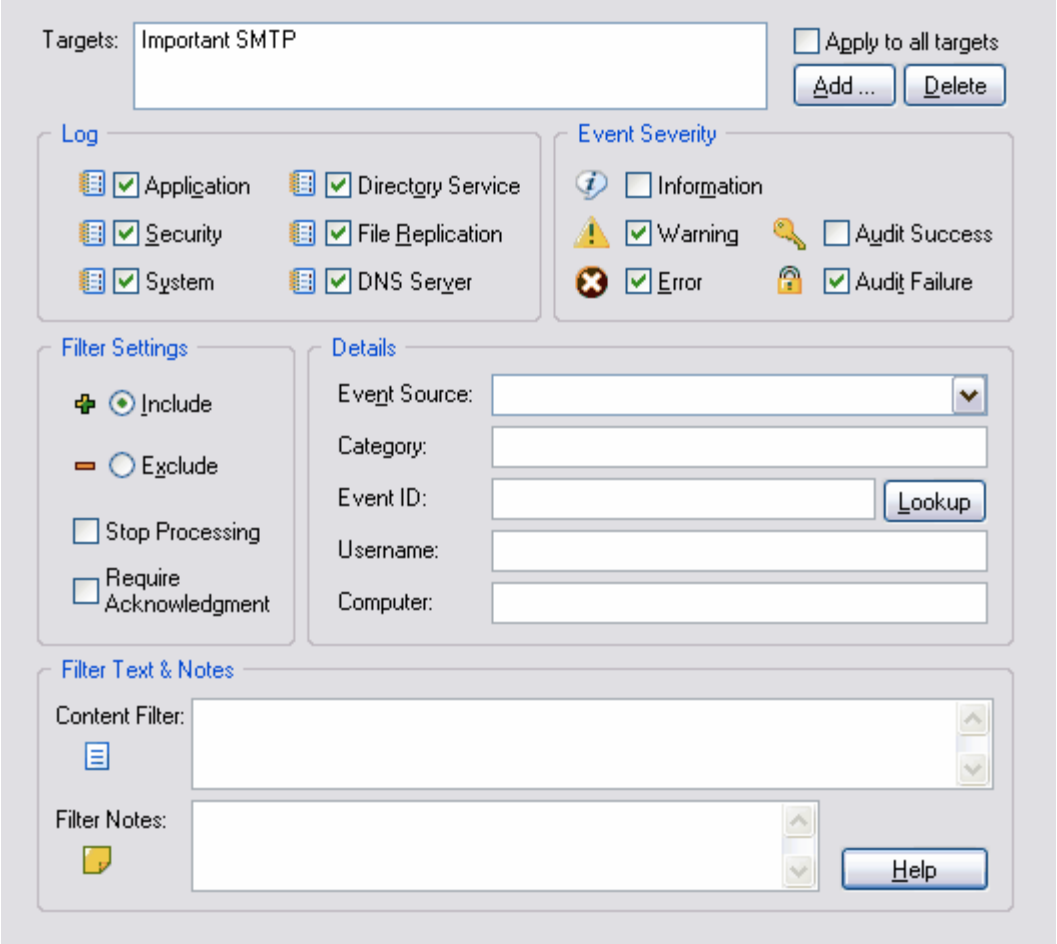


Figure 16

If you prefer to create a filter solely to forward events created by the recurring event filter, then use the settings shown below:

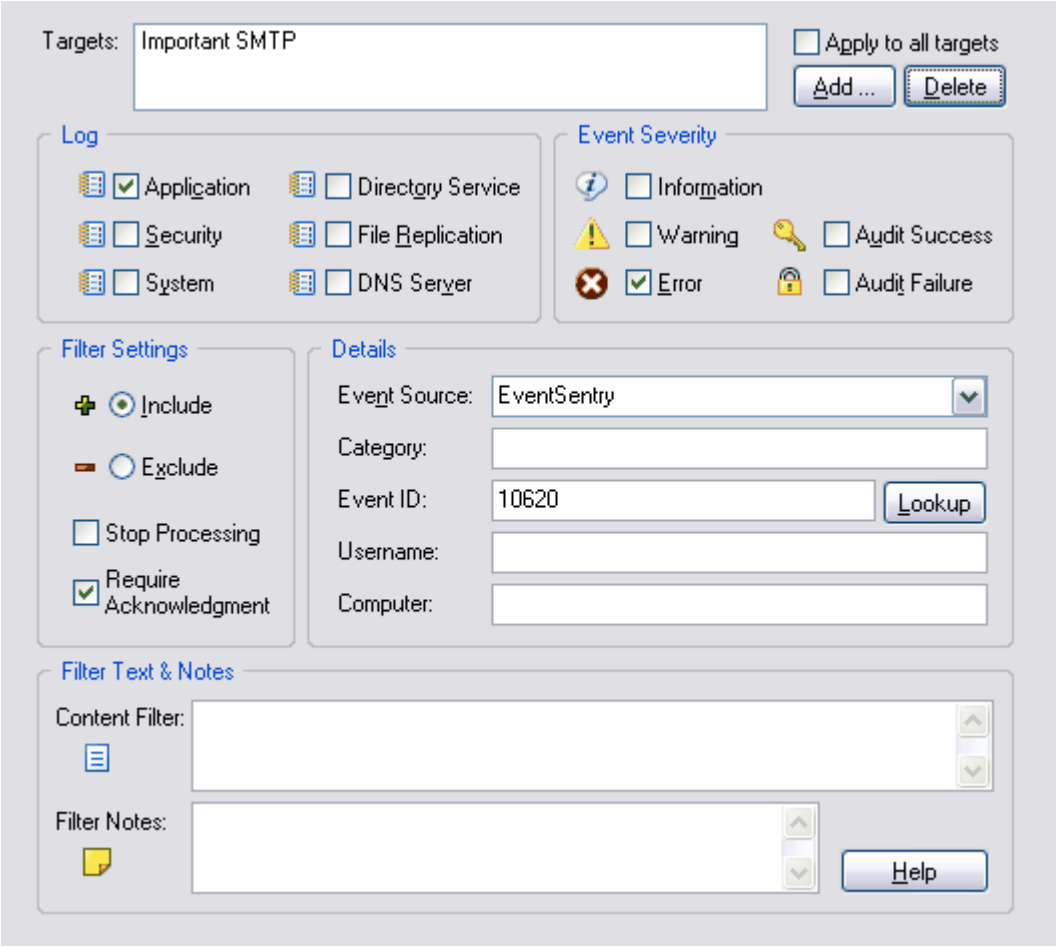


Figure 17

And that's it! With this setup you will always know immediately if your backup didn't run or only ran partially, without lifting a finger. If the backup didn't work then you will get an email in your inbox first thing in the morning.

5.2 Log Backup Activity To Database

In addition to knowing that the backup didn't work, it is still desirable to log backup activity to some sort of log file, e.g. to a database. That way the backup history will still be available even when the event log is cleared.

As of version 2.70, EventSentry supports the following databases:

- **Microsoft Access:** This database is only recommended when you have a single host or a very small number of computers (e.g. 1-3 computers) in your network.
- **Microsoft SQL Server:** This is the recommended database for small, medium and large networks and has thus far shown the best performance with the web reports.
- **MySQL:** This database is recommended when you have a small or medium size network and do not have Microsoft SQL Server available.
- **Oracle:** We only recommend Oracle for customers who already have an Oracle database server they need to integrate EventSentry with.

Since this guide focuses only on one computer we will use a Microsoft Access database for our

example. In order to log all backup (or all event log activity for that matter) to the database you will need to do the following:

1. Install the EventSentry Access database (automatically done during installation)
2. Create a database notification target (automatically done during installation)
3. Create a filter forwarding NTBackup events to the database
4. Optionally create a custom Access query
5. Setup the web reports using IIS

5.2.1 Install Access database and setup notification

When installing EventSentry you have option of installing the EventSentry Access database. This database already contains all required tables, queries and forms.

When running the EventSentry installation, make sure that you select the option **Install Sample MS Access Database** (under Database Features) so that the MS Access database **EventSentry.mdb** is copied to the installation directory (e.g. C:\Program Files\EventSentry).

If a database notification for the MS Access database has not been created then you will need to create it manually. Right-click the **Notifications (Targets)** container and select **Add** from the menu to add a new notification. Give the new notification a descriptive name (e.g. **Access Database**) and hit ENTER. Then select the **ODBC** tab (all database logging is done through ODBC with EventSentry) and click the **Create** button in the "Connection String" area.

Configure the connection string according to the options below. This includes setting the Database Provider to **MS Access** and browsing for the **EventSentry.mdb** file. Specifying a username and password is usually not necessary.

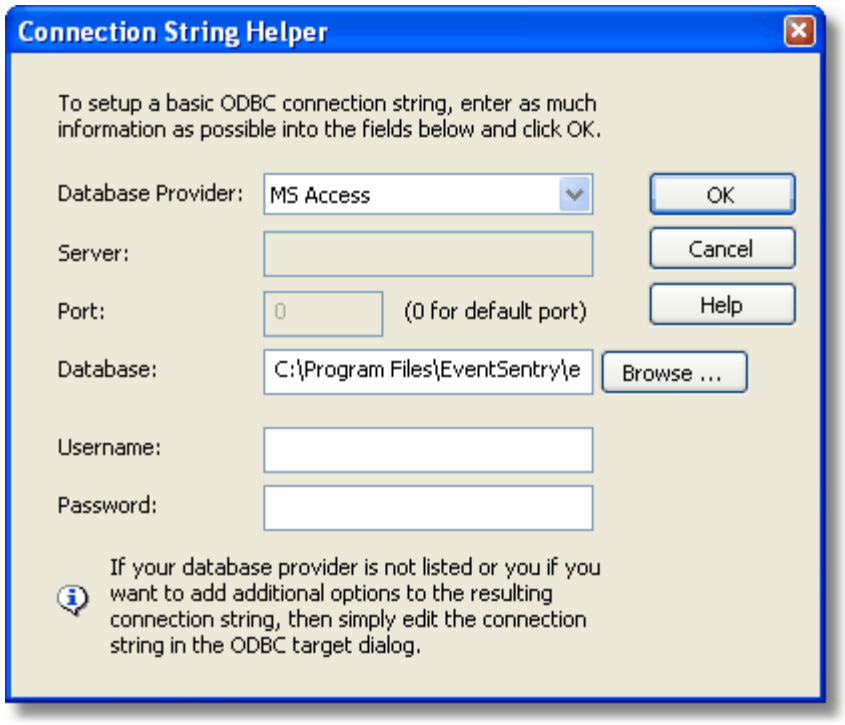


Figure 18

The setup for this notification is now complete and we can setup a filter to forward NTBackup events to the database.

5.2.2 Creating a filter

Add an additional filter by right-clicking a previously created filter or by right-clicking the "Filters" container and selecting **Add Filter**. Enter a descriptive name (e.g. "Log Backup") and hit ENTER.

Then, configure the general filter details as shown below:

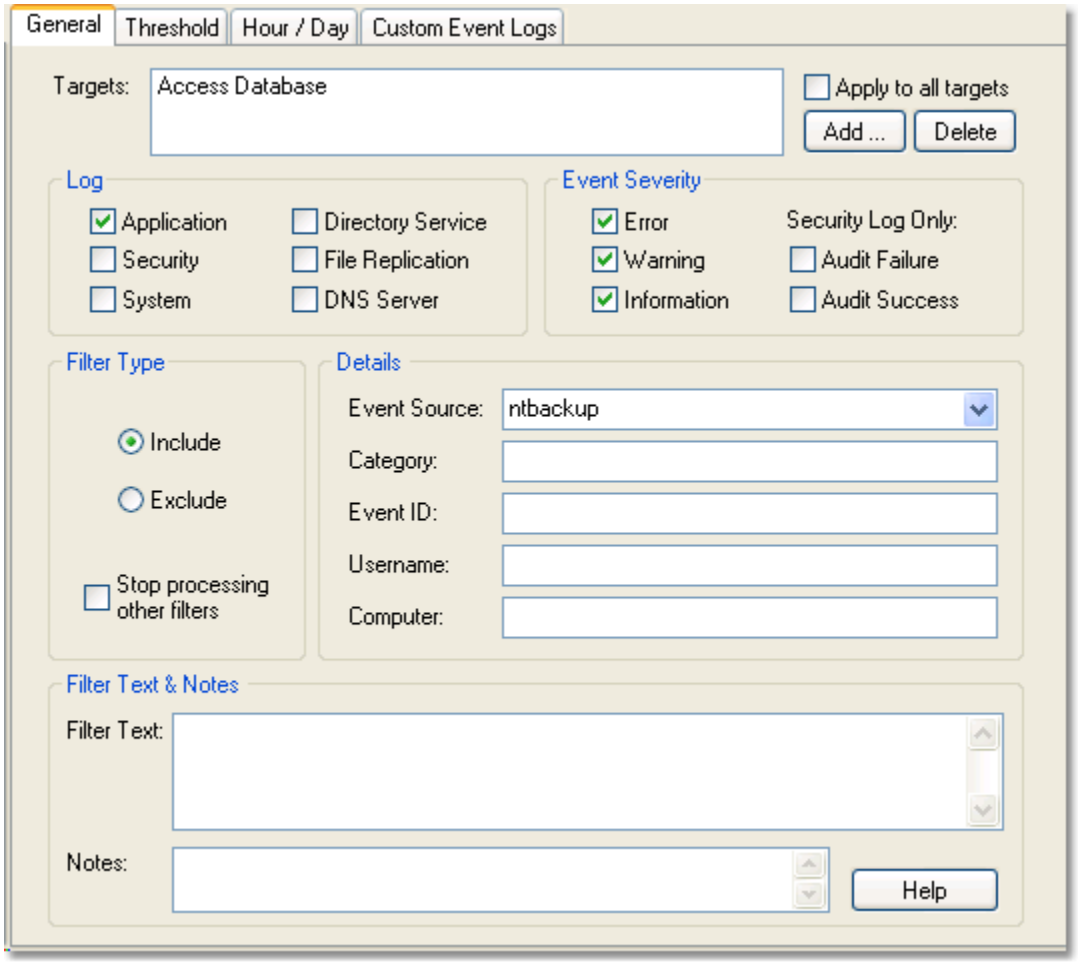


Figure 19

If you want to log more than just NTBackup events then you can simply leave the **Event Source** field empty and optionally select all other logs (Security, System, etc.) and severities (Audit Failure, Audit Success). The filter in figure 19 will forward all events logged by the **NTBackup** source to the **Access Database** notification.

5.2.3 Creating a custom Access Query

If you have Microsoft Access installed then you can simply open the EventSentry access database (EventSentry.mdb) and look at the **Event Log** query to see a history of all backup activity.

If you are logging more than just the **NTBackup** events to the database then you can also copy the **Event Log** query to create a new query that will only show **NTBackup** events. Open the EventSentry access database and click the **Queries** button:

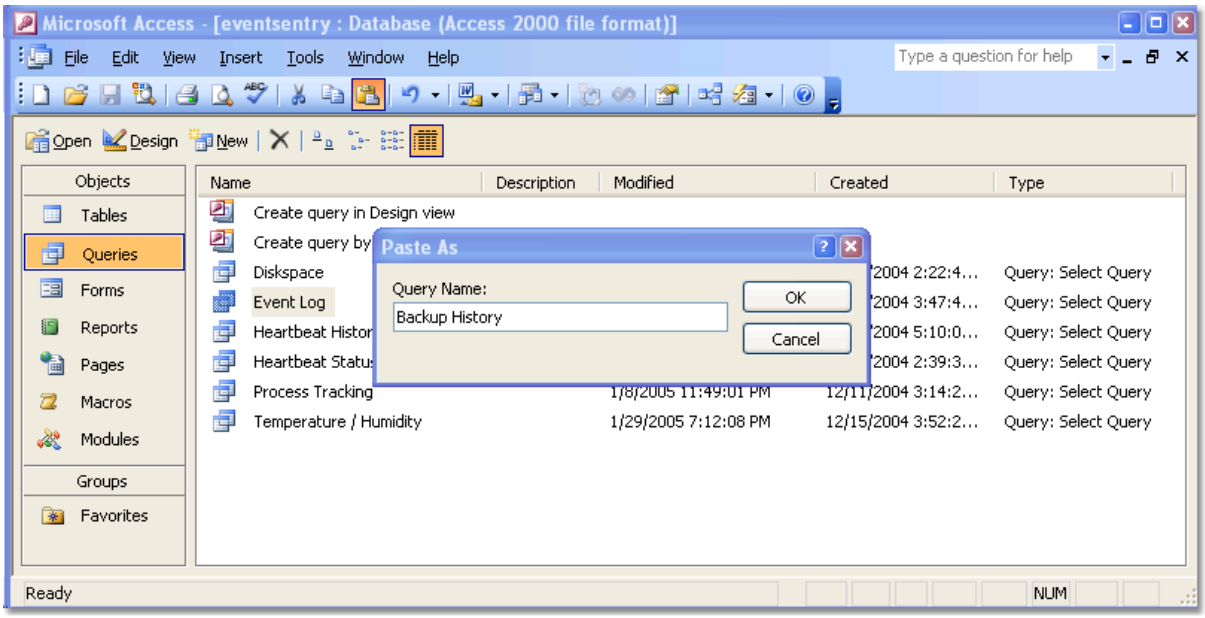


Figure 20

Select the **Event Log** query and click the **copy** button and then the **paste** button on the toolbar. Enter a descriptive name (e.g. **Backup History)** when prompted for a name of the new query.

To change the query select it and click the **Design** button on the toolbar. In the design editor, specify **NTBackup** for the criteria and click the save button as shown below:

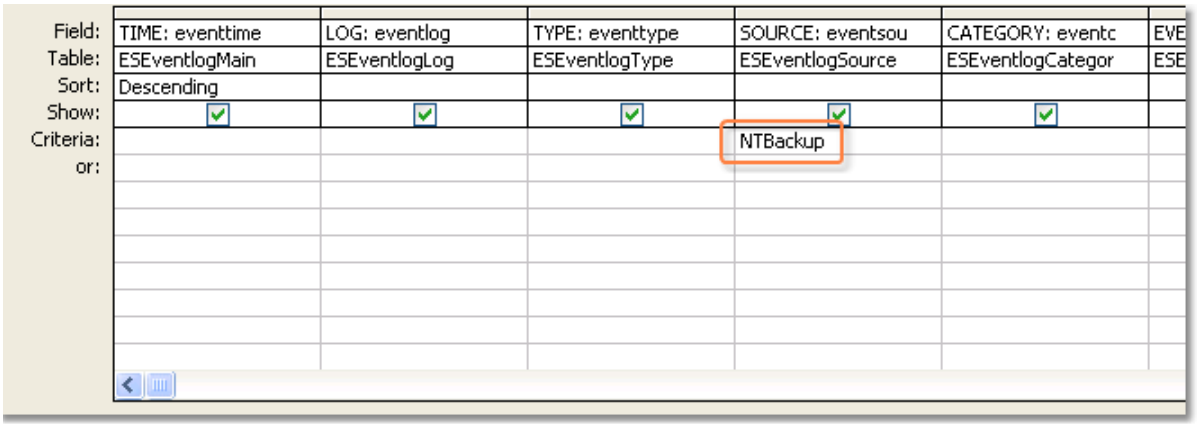


Figure 21

and the resulting query will only show events from the **NTBackup** source.

5.2.4 Setting up the web reports

If you don't have Microsoft Access installed or if you would like to view the backup history through a web browser (which is definitely more convenient) then you will need to setup the EventSentry web reports. The web reports files (ASP files) need to be installed and configured on a computer running **Internet Information Server**.

Explaining how to setup the web reports is beyond the scope of this document, but the EventSentry manual dedicated an entire chapter on how to install and configure the web reports.

The basic steps involved are:

1. Run the EventSentry setup on the machine where IIS is installed. If you have not yet installed IIS, then run the EventSentry setup **after** you installed IIS. The EventSentry setup will automatically create a new properly configured virtual directory.
2. Create a system DSN that will point to the Access database.
3. Edit the **WebReportsConfig.xml** configuration file and point it to the previously created system DSN.

5.3 Email Detailed Backup Log

In addition to being notified that the backup has failed, it is often useful to examine the log output from NTBackup right away. You can configure EventSentry to automatically email you the detailed log output from NTBackup when the backup does not complete successfully.

1. **Create Notification Script:** The script is responsible for finding the latest backup log file and emailing it using **blat.exe**.
2. **Create Process target inside EventSentry:** The target will point to the script file created above and be used in a filter.
3. **Create Filter:** Create a filter that will look for the 10620 event generated when the backup doesn't complete.

5.3.1 Create Notification Script

We will use EventSentry's ability to trigger any process from an event to trigger our VBScript below to emails us the latest log file.

Blat
Blat.exe is a public domain, freeware command-line email utility, similar to sendmail on Linux/Unix (though it is only an email client, it is not a MTA etc.). Download blat from http://www.blat.net, and we recommend that you save the blat.exe file in the same directory where you are going to put the .vbs script later on, for example `C:\Batch\blat.exe`.

Visual Basic Script File
Using any text editor, create a new file and immediately save it in a folder on your server, for example:

```
c:\batch\es_email_ntbackup_log.vbs
```

Again, we recommend that you save this file in the same folder where you saved blat.exe. Then, paste the formatted output below into the file and modify all the values shown in bold. This is imperative, since you need to specify your email configuration. All required values are explained below:

SmtpServer	The host name or IP address of your SMTP server
EmailSender	The email address that will appear as the sender
EmailRecipient	The email address of the recipient
LogFolder	All log files from NTBackup are written to this directory, but you might have to replace **Administrator** with the actual username under which your NTBackup is being scheduled.
Compress	Set this value to one if you want to compress the log file (highly recommended) and if you have a zip application installed that supports command-line based compression
CompressExe	The path to the command-line .exe compression utility. If you do not use Winzip then you will have to also change the command-line argument below.
EmailClient	Full path to the blat.exe file

```
' Author: NETIKUS.NET ltd
' Date:   9/27/2006
```

```
' Title:  Email NTBackup Log file using blat.exe
' Description: Emails the latest ntbackup.exe log file using blat.exe, and
optionally compresses the attachment

Dim SmtpServer
Dim EmailSender, EmailRecipient, EmailSubject

Dim aFile
Dim fNewest, fNewestZipped
Dim oFolder
Dim LogFolder
Dim numFiles

Dim Compress, CompressExe
Dim EmailClient

' Set your preferences here *
SmtpServer        = "emailserver-hostname"
EmailSender       = "SERVER@yourdomain.com"
EmailRecipient    = "youremail@yourdomain.com"
EmailSubject      = "NTBackup Log File"
LogFolder         = "C:\Documents and Settings\Administrator\Local Settings
\Application Data\Microsoft\Windows NT\NTBackup\data"
Compress          = 0   ' Set to 1 if you have Winzip installed
CompressExe       = "C:\Program Files\Winzip\wzzip"
EmailClient       = "C:\Batch\blat.exe"

Set WshShell      = WScript.CreateObject("WScript.Shell")
Set oFolder       = CreateObject("Scripting.FileSystemObject").GetFolder
(LogFolder)

numFiles = 0

' Find log file with latest time stamp
For Each aFile In oFolder.Files
      If Right(aFile.Name, 4) = ".log" Then
            numFiles = numFiles + 1

            If fNewest = "" or fNewest = null Then
                  Set fNewest = aFile
            Else
                  If fNewest.DateLastModified < aFile.DateLastModified Then
                        Set fNewest = aFile
                  End If
            End If
      End If
Next

' Only compress if configured to do so
If Compress = 1 Then
      fNewestZipped = fNewest & ".zip"
Else
      fNewestZipped = fNewest
End If

' Email log file
If numFiles > 0 Then
      If Compress = 1 Then
            CommandLine = """" & CompressExe & """ -u """ & fNewestZipped &
""" """ & LogFolder & "\" & fNewest.Name & """"
```

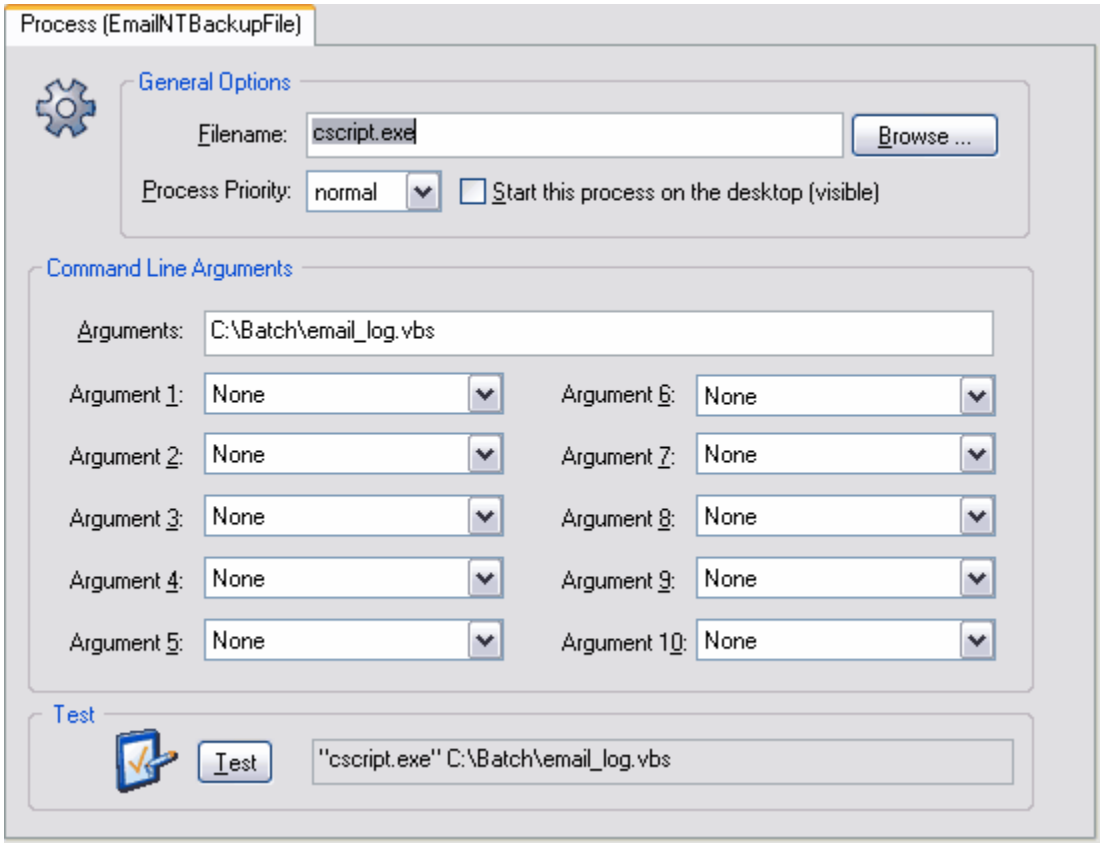
```
            WshShell.Run CommandLine, 3, 1
        End If

        WshShell.Run EmailClient & " - -body ""NTBackup Log File"" -attach
""" & fNewestZipped & """ -server " & SmtpServer & " -f " & EmailSender & "
-subject " & chr(34) & EmailSubject & chr(34) & " -to " & EmailRecipient,
3, 1
End If
```

5.3.2 Create Process Target

You will need to setup a process target (notification) before EventSentry can execute any system process. This process target can then be referenced by one or more filters.

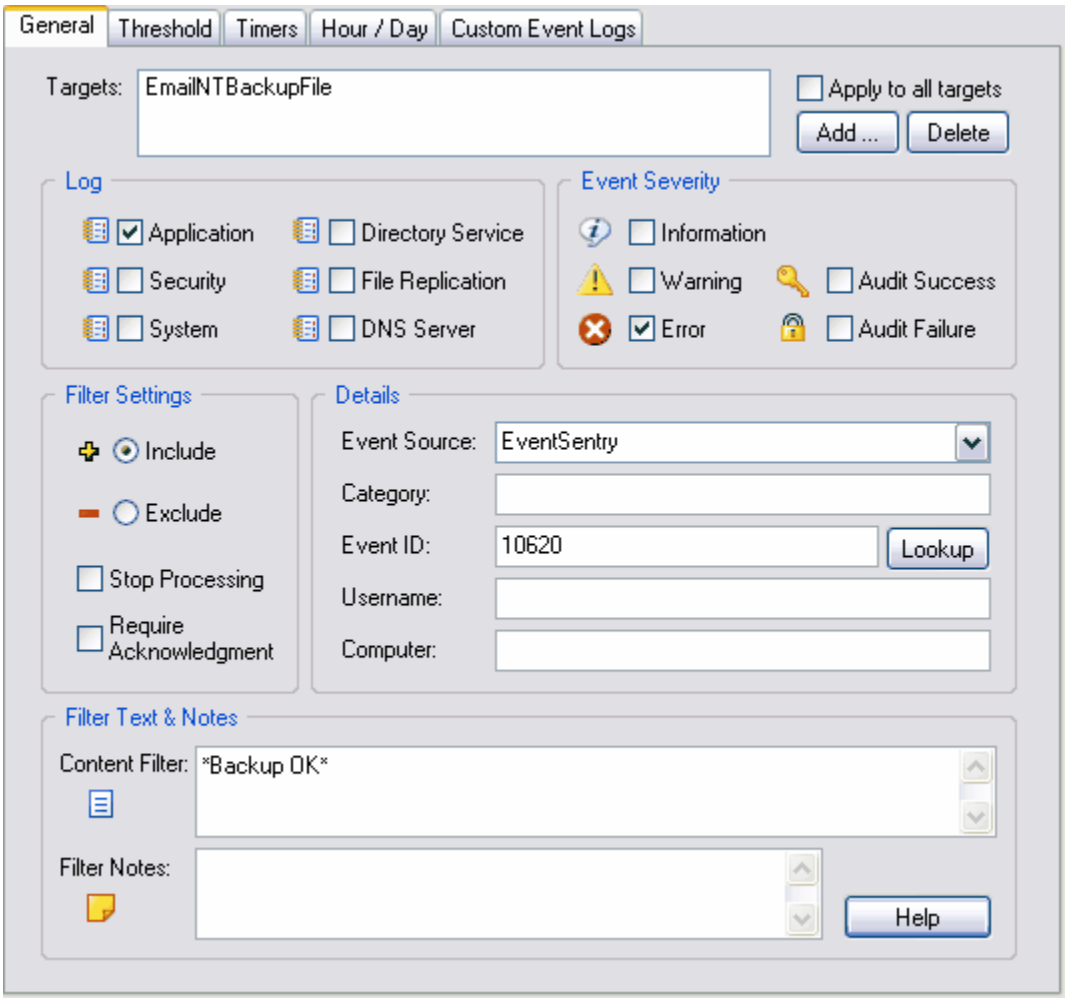
Right-click the Notifications (Targets) container and select **Add**. Then, specify a descriptive name (e.g. "**EmailNTBackupFile**") and configure the process target to point to the .vbs file as shown in the dialog below:



It is important to understand that we need cscript.exe to execute the our VBS file, since cscript.exe is the command-line interpreter for Visual Basic Script. If you did not save the VBS file in the folder shown above then you will have to adapt the configuration accordingly.

5.3.3 Create Filter to execute Process Target

Last but not least we need to trigger the script when our backup has indeed failed, that is, when our recurring event filter logs the famous **10620** event. As such, create a new filter in the NTBackup filter package you created earlier, and configure it similar to the output shown below:



This will cause EventSentry to trigger our script, which emails us the latest .log file generated by NTBackup.

5.4 Configure Weekly Summary

If you would like the added security of knowing that the backup ran successfully every day, without getting an email every single day, then you can receive a summary email with all backup events once a week.

All you need to do to receive this summary email is to add one (summary) filter that will collect and send emails on a weekly basis. Add an additional filter by right-clicking a previously created filter (or by right-clicking the "Filters" container) and selecting **Add Filter**. Enter a descriptive name (e.g. "Backup Summary") and hit ENTER.

Configure the general filter properties as shown below:

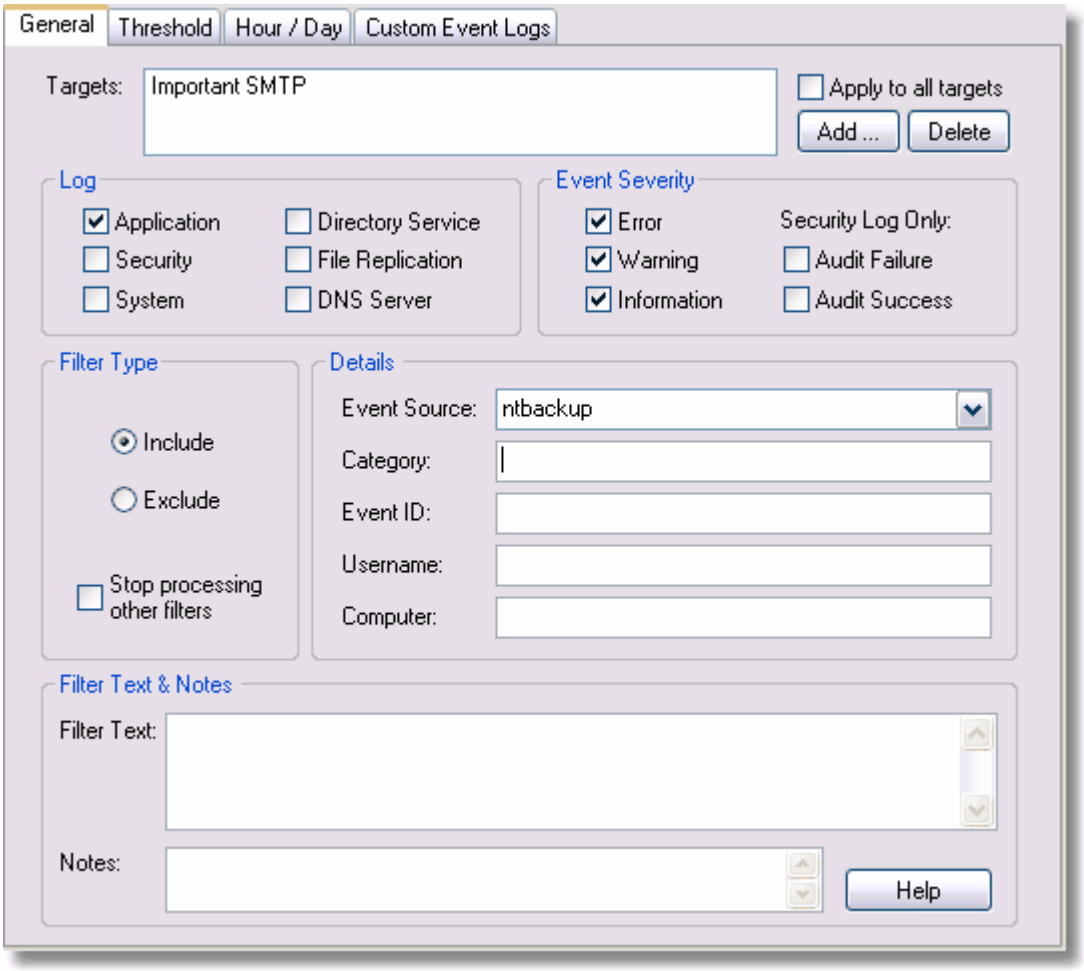


Figure 22

Then, click the **Hour / Day** tab and set the summary schedule, and don't forget to set the schedule type to **Summary** as well. The schedule shown below will send a summary email every Monday at 7AM in the morning:

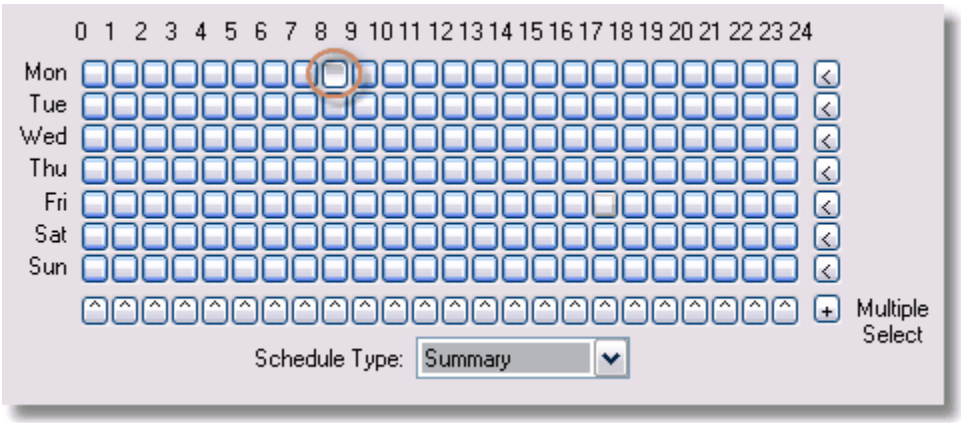


Figure 23

When the schedule type is set to **Summary**, then EventSentry will queue events during inactive hours (blue buttons) rather than ignoring them. The events are queued until the first active hour (white

button) is reached. It is then when all the queued events are sent out. Please see the EventSentry manual for more information on summary filters.

EventSentry summary filters are useful for a variety of scenarios, not just for receiving a backup summary. Please note however that EventSentry requires you to create a unique SMTP notification for each summary notification filter that you create. This is due to the way how summary notifications are processed by the EventSentry agent.