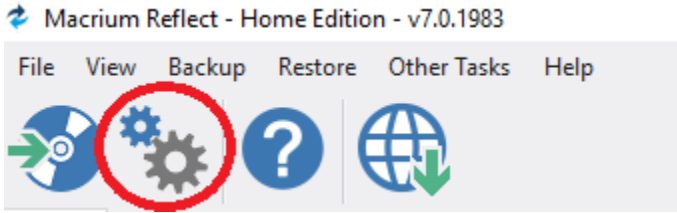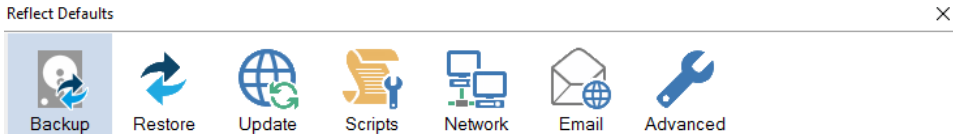# Macrium Reflect default settings

The Macrium Reflect default settings allow you to set the default values used when creating a new backup definition.

1. To edit default settings from the main toolbar click the 'Cog' icon as shown below.

2. Select the area you want to amend.

3. When you are happy with your selections click **OK**.

To change existing definition, right click on the definition and select 'Advanced Properties'.

- Default Backup settings
- Default Restore settings
- Update Settings
- Scripts
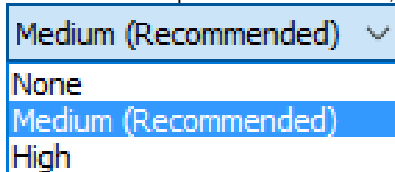- Email Settings
- Events
- Advanced

Default Backup settings

Compression is used to reduce the file size of the completed backup.

These options set the defaults for a new backup.
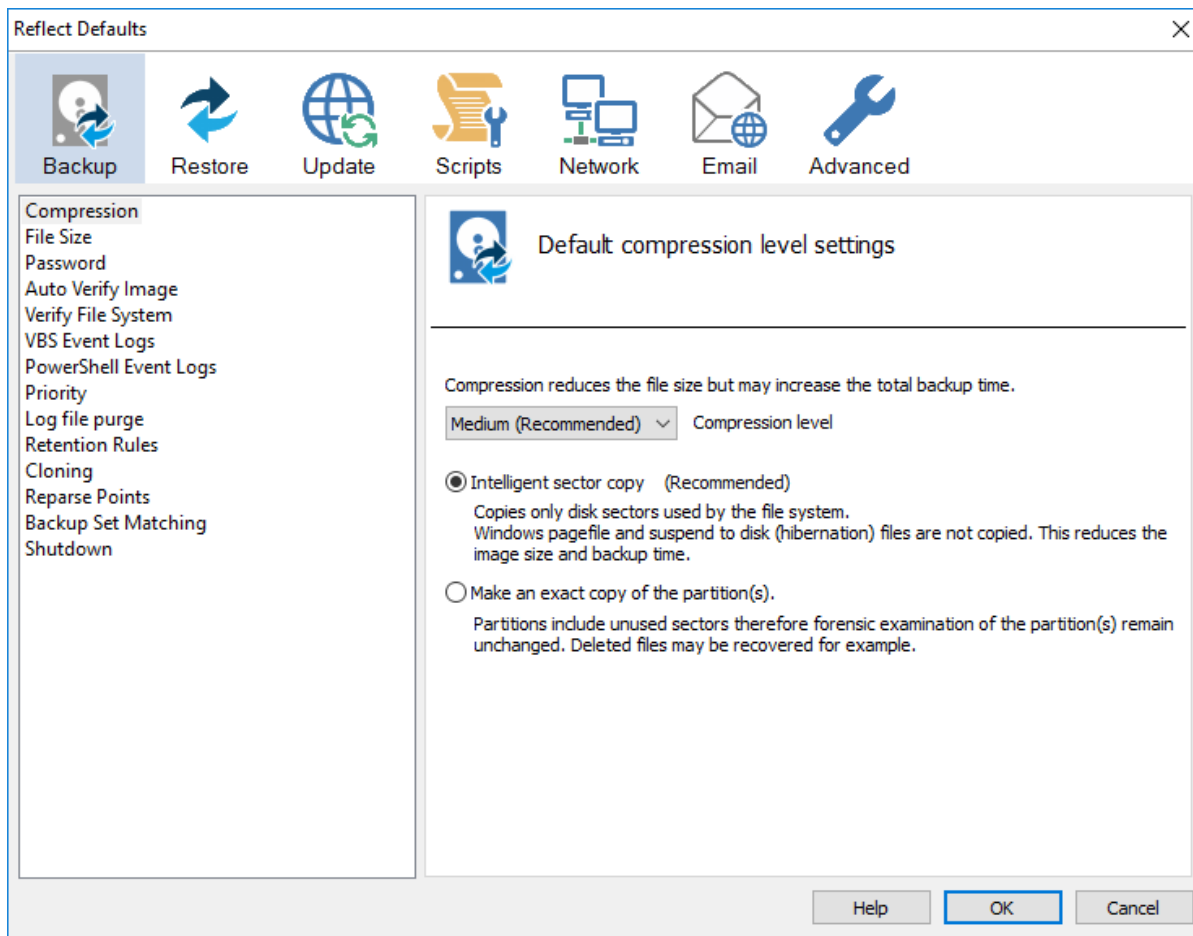
- Three levels of compression can be chosen, the higher the compression level the smaller the backup file will be:


Reducing the file size may increase the total backup time.

Compression allows you to chose an Intelligent sector copy or a Forensic copy of the partitions.

Option	Description
Intelligent Sector Copy	Only backup the sectors that are being used by data on the disk. Pagefile (pagefile.sys) and hibernation (hiberfil.sys) will also be excluded. This reduces the time it takes for the backup to complete.
Forensic Copy	Backup every sector.

File size options are used to select between automatic or fixed file size for an image.

These options set the defaults for a new backup.
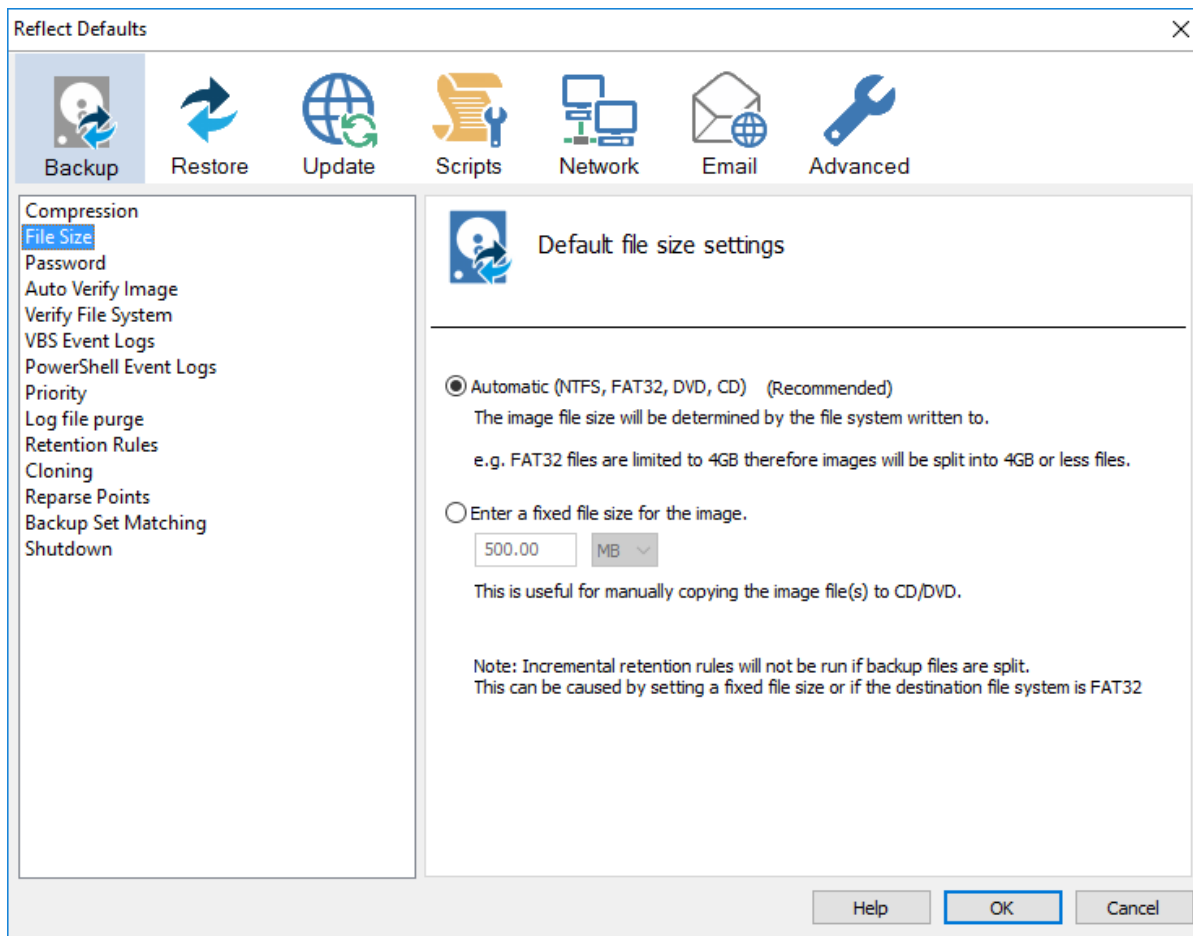
Incremental Retention Rules will not be run if backup files are split. This can be caused by setting a fixed size or if the destination file system is FAT32.

Option	Description
Automatic (Recommended)	Let the system decide on how large the images are going to be created dependent on file system (NTFS, FAT32, DVD, CD) e.g FAT32 files are limited to 4GB therefore images are going to be split into 4GB or less files.
Fixed file size	Create Images that will be split into many fixed size files. This is useful when copying Image files to CD/DVDs.

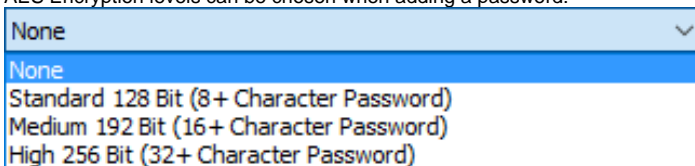**Password** option is used to enable password protection.

These options set the defaults for a new backup.
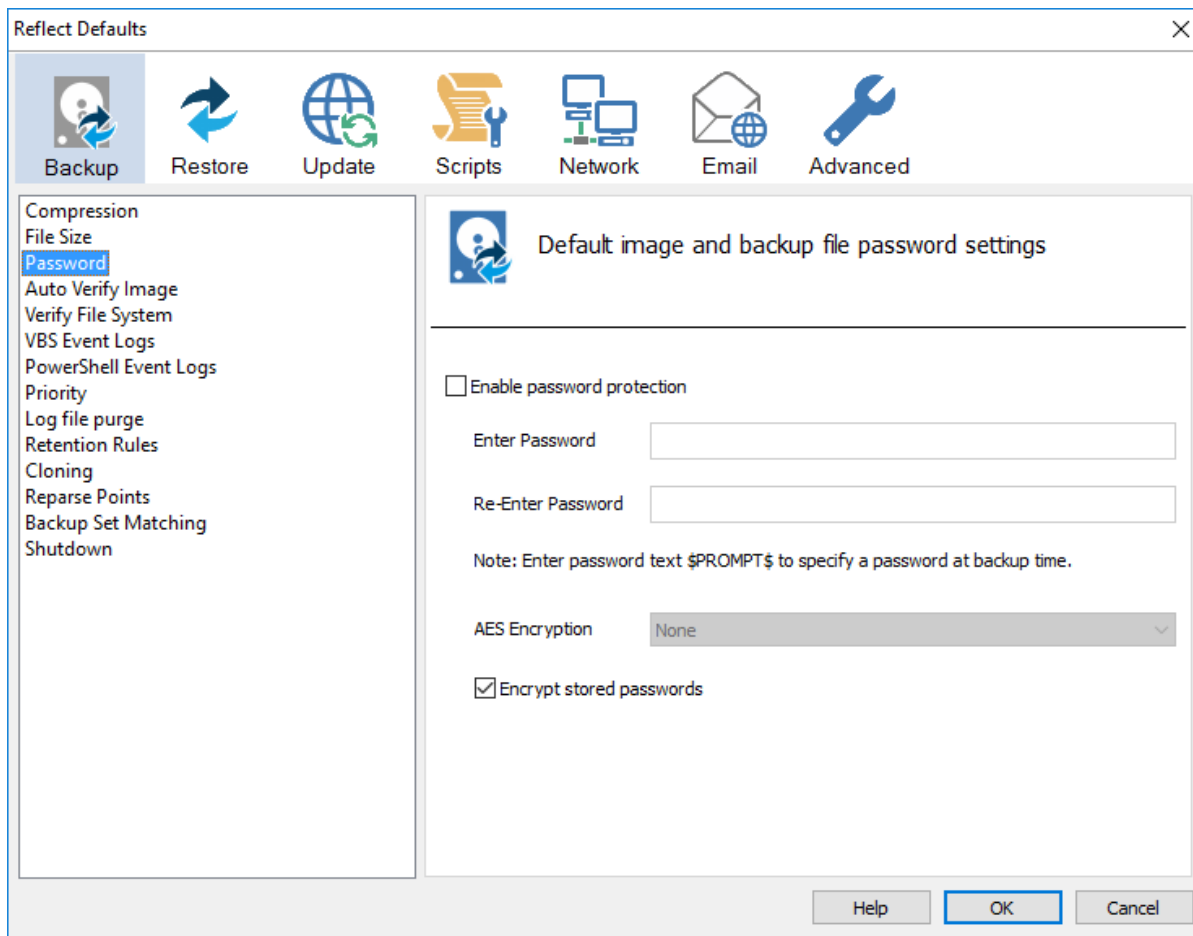
- With the **Password** option turned on all the images created will require a password before they can be browsed or restored.

Enter password text $PROMPT$ to specify a password at backup time.

- AES Encryption levels can be chosen when adding a password:

AES 256 bit encryption is the highest level of encryption available.

Option	Description
Enable password protection	Enables protection of your image files.
Encrypt stored passwords	Passwords saved in your backup definition will be encrypted.

Auto Verify Image allows automatic verification of images after they are created.

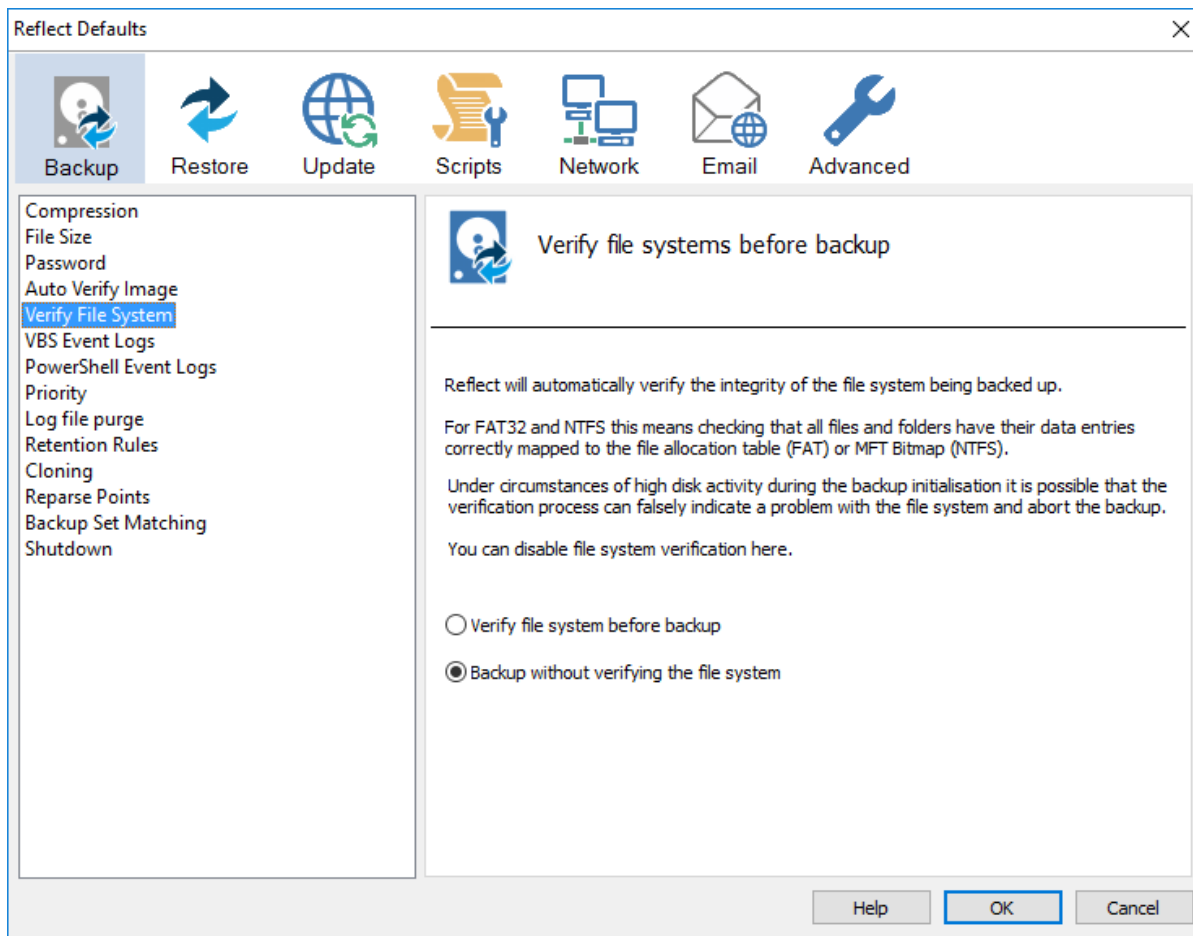These options set the defaults for a new backup.

Reflect Defaults ✕

| Backup | Restore | Update | Scripts | Network | Email | Advanced |

Compression
File Size
Password
Auto Verify Image
Verify File System
VBS Event Logs
PowerShell Event Logs
Priority
Log file purge
Retention Rules
Cloning
Reparse Points
Backup Set Matching
Shutdown

Default verify image or backup file creation settings

Select to automatically verify the integrity of your image or backup file directly after it is created. If files are split then each file is independantly verified.

Note: This may add a significant amount of time to the backup process.

☐ Verify image or backup file directly after creation

Help   OK   Cancel

This can add a significant amount of time to the backup process.

Option	Description
Verify image	Images will be verified automatically when the backup completes.

For more information on image verification please see Understanding Image Verification Failures

**Verify File System** is used to check the integrity of the file system before a backup.

These options set the defaults for a new backup.

Reflect Defaults ✕

Backup	Restore	Update	Scripts	Network	Email	Advanced

Compression
File Size
Password
Auto Verify Image
Verify File System
VBS Event Logs
PowerShell Event Logs
Priority
Log file purge
Retention Rules
Cloning
Reparse Points
Backup Set Matching
Shutdown

Verify file systems before backup

Reflect will automatically verify the integrity of the file system being backed up.

For FAT32 and NTFS this means checking that all files and folders have their data entries correctly mapped to the file allocation table (FAT) or MFT Bitmap (NTFS).

Under circumstances of high disk activity during the backup initialisation it is possible that the verification process can falsely indicate a problem with the file system and abort the backup.

You can disable file system verification here.

○ Verify file system before backup

◉ Backup without verifying the file system

Help	OK	Cancel

- Reflect will automatically verify the integrity of the file system being backed up.
  For FAT32 and NTFS this means checking that all files and folders have their data entries correctly mapped to the file allocation table (FAT) or MFT Bitmap (NTFS).

Option	Description
Verify file system before backup	Verify the file system on your disk prior to the backup; this will increase the time taken to complete your backup.
Backup without verifying the file system	Backup without verifying, thus making your backup a little quicker.

Priority is used to change the CPU priority for images and backups.
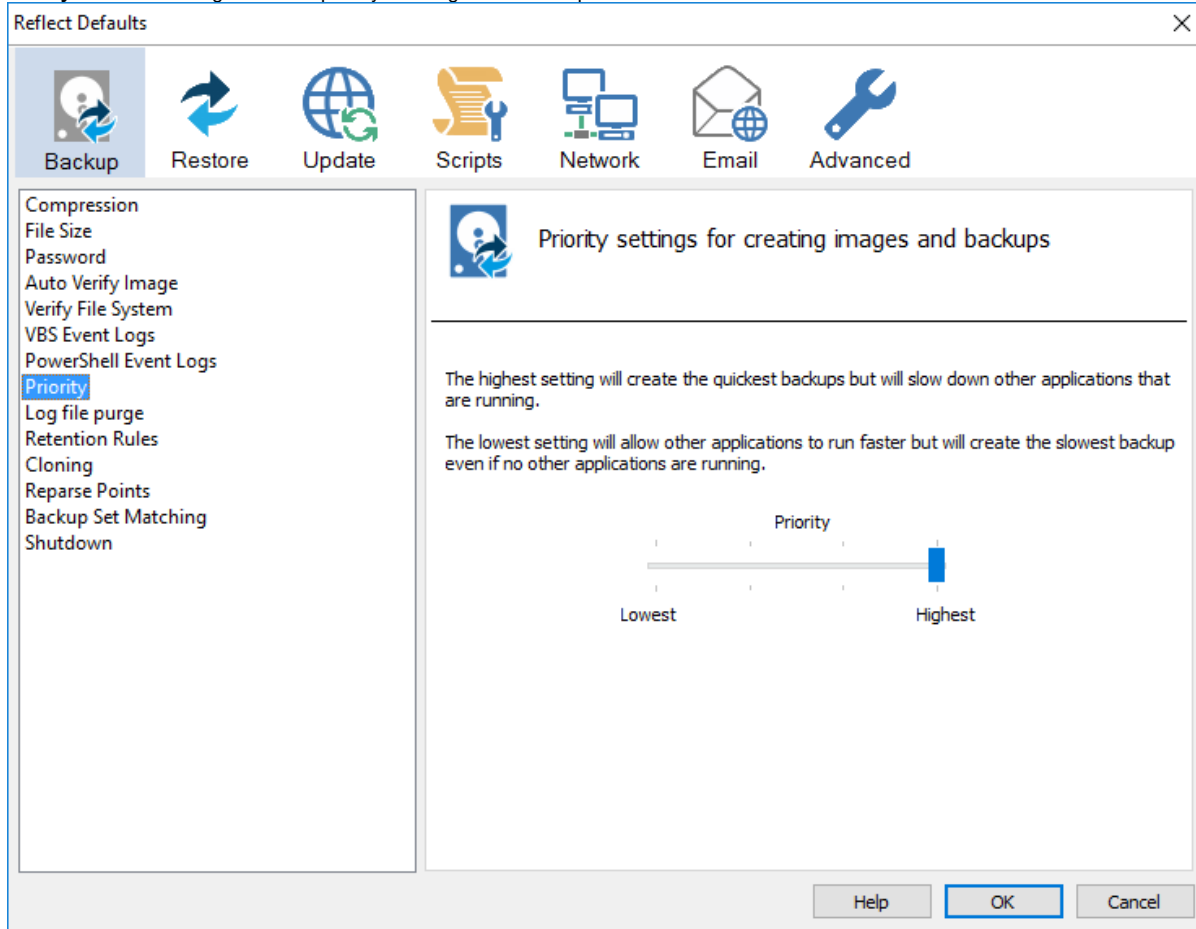
Option	Description
Highest	Your backup will take less time to complete and may slow down other applications.
Lowest	Your backup may take longer to complete but it will have less effect on other running applications.

**Log file purge** option enables you to manage the retention of your backup logs.
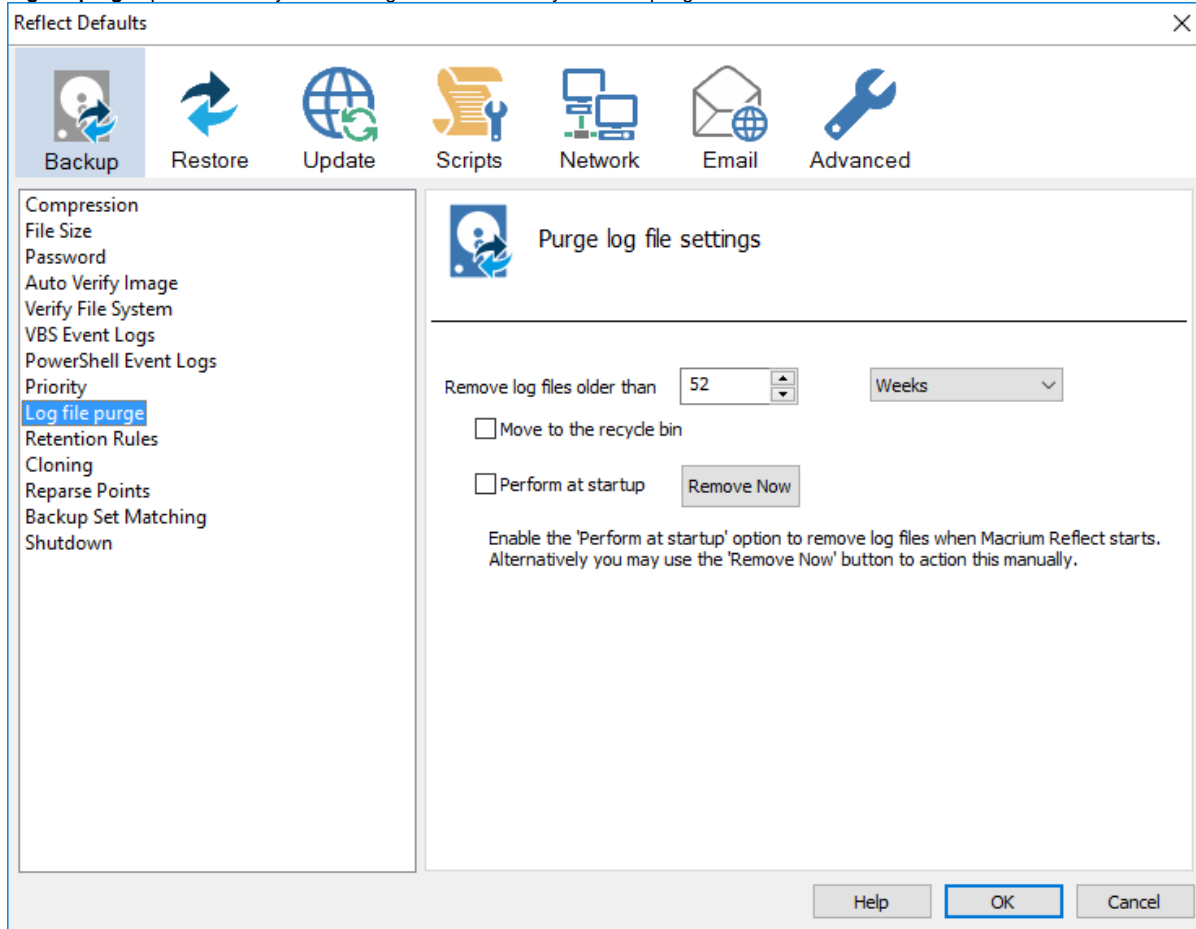
Option	Description
Remove log files	Choose number of days of weeks
Move to Recycle Bin	Deleted logs will be moved to the Recycle Bin.
Perform at startup	Logs will be deleted when Reflect starts up.

You can press the **Remove Now** button to remove the logs manually at any time.

Retention Rules are used to define how long you want to keep different backup types (Incremental, Differential and Full).

These options set the defaults for a new backup.
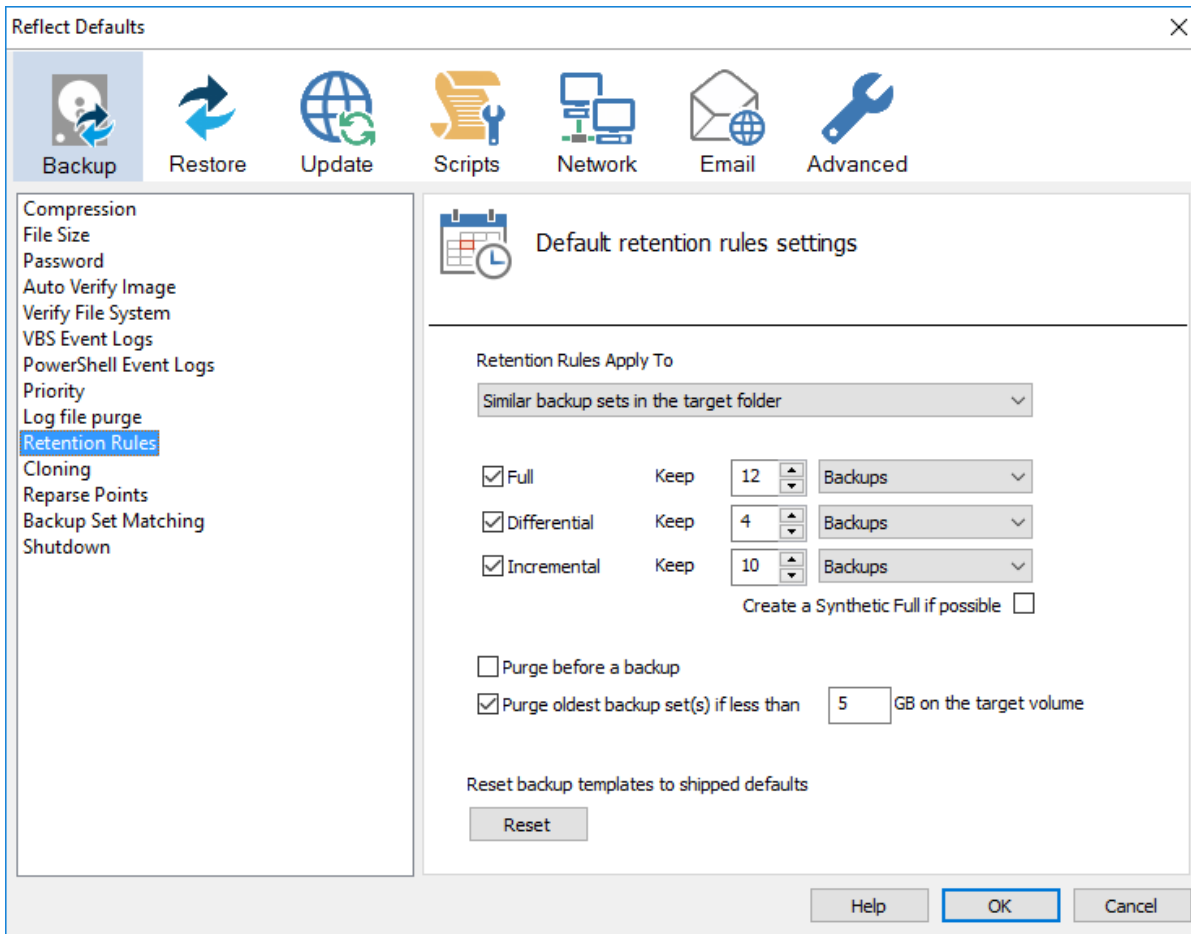
The new Macrium Reflect retention rules provide a powerful and flexible way to manage the lifetime and storage space used by your backups.

Choose how backups are matched and retention rules are applied to the target folder

Retention rules are applied to the target folder of the backup by selecting one of two options:
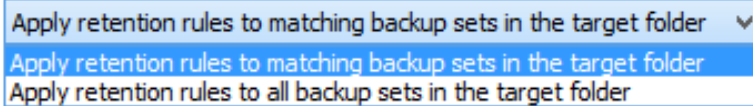
1. **Apply retention rules to matching backup sets in the target folder.**

   Disk Images are purged if they contain **exactly the same Partitions** as the current Image. Partitions are identified using the unique **Disk ID** stored in sector 0 of the disk and the **Partition sector offset**.

   Note: For GPT disks the unique GPT disk GUID is used instead of the Disk ID

   For **File and Folder** backups retention rules are applied according to the '**Backup Set Matching**' option select in the **'Advanced Properties'** for this backup.

2. **Apply retention rules to all backup sets in the target folder.** All backup sets in the target folder of the same type (Disk Image or File and Folder) are purged according the retention rules.

   Note: This option uses the same logic as Macrium Reflect v5

Select the age or number of backup types that you wish to keep

Option	Description
Full	When deleting Full backups all linked incremental and Differential backups in the same backup chain (set) are also deleted This operation will delete the entire backup set.
Differential	When deleting Differential backups all linked incremental backups in the same backup chain (set) are also deleted.

Incremental	When deleting Incremental backups the integrity of the backup set is maintained by ensuring that the chain is never broken. This is achieved by merging older Incremental backups when required.
	
	 In the example below, before retention, there is **1 Full backup, 1 Differential backup** and **6 Incremental** backups. The retention rules are set to **retain 4 incremental** backups. After retention, the most recent 4 incremental backups are retained. **Deleting the oldest 2 incrementals would cause the backup chain to be invalid** as the oldest retained incremental requires the previous 2 incremental backups to complete the chain. To ensure backup integrity the **2 older incremental backups are consolidated** with it to create a new incremental backup.
	
	 `F = Full`
	`D = Differential`
	`I = Incremental`

Create a Synthetic Full if possible	When purging Incremental backups, if the backup set **only contains a Full backup followed by Incremental backups**, then this option causes the Full backup to be 'rolled forward' to **create a Synthetic Full backup**. This is also known as **Incremental Forever.**
Run the purge before the backup	Select this option to run the retention rules before the current backup. **Note:** in Macrium Reflect v5 the current backup set wasn't included in the purge calculation when purging before the current backup. In v6 the current backup set **IS** included. This means that if you set the retention count to 1 Full backup then all of your backups will be deleted and a new Full backup created.
Delete oldest backup set(s) if less than n GB	Automatically remove the oldest backup set(s) in the target folder if the free space on the drive drops below the GB threshold. **Note:** The free space threshold is actioned dynamically. If the free space available drops below the threshold then the running backup is temporarily paused while older backup sets are purged.

Within the Incremental description, the example table:

M	T	W	T	F			M	T	W	T	F			M	T	W
F							D	I	I	I	I			F	I	I
								--	->	I						

Cloning

These options set the defaults for a new clone.

| Option	Description

Perform an Intelligent Sector Copy	Only backup the sectors that are being used by data on the disk. Pagefile (pagefile.sys) and hibernation (hiberfil.sys) will also be excluded. This reduces the time it takes for the clone to complete.		
	Option	**Description**	
	Verify File System	Reflect will verify the integrity of your file system; Verification check that all files and folders data entries are correctly mapped to the file allocation table (FAT) or MFT Bitmap (NTFS)	
	Rapid Delta Clone	As with Rapid Delta Restore (RDR) the concept of RDR has been something that has been thought about for quite some time here at Macrium Software. We wanted to build a clone solution that would effectively and rapidly copy only the differences between the source and target file systems. The advantage of this is obvious, RDC offers similar a performance increase as an Incremental disk image offers over a Full image and enables regular clones to be a viable and fast DR solution. **How does it work?** The NTFS file system resident on the clone source is compared with file system on the target disk. The two file systems are first verified that they originated from the same format command and then the target NTFS file system structures are analyzed for differences. All the NTFS file system structures are copied to the target disk and any that do not exist or have been modified on the target disk cause the data records for each NTFS file or object to be copied as well. The result is an 'Incremental' clone applying only file system changes detected between the source and the target. **Note:** RDC works with NTFS file systems only. All other file systems will perform a full clone **Note:** RDC is not available when shrinking partitions during a clone.	
	Enable SSD TRIM	This features provides automated SSD optimization resulting enhanced SSD performance and longevity. Writing to an unused block is much quicker than an in-use block as it avoids both the slow erase operation and the read-modify-write cycle. This results an increase of both the lifetime and the performance of the device. It is effective for all windows operating systems, even those that support SSD trim natively as the file system driver can only TRIM blocks on de-allocation; it cannot TRIM blocks written by another process. It is also effective for USB attached SSDs.	
Perform a Forensic Sector Copy	Backup every sector. This can add a significant amount of time to the backup process.		

Reparse Points options are used to include or exclude Reparse Points from a backup.

These options set the defaults for a new backup.
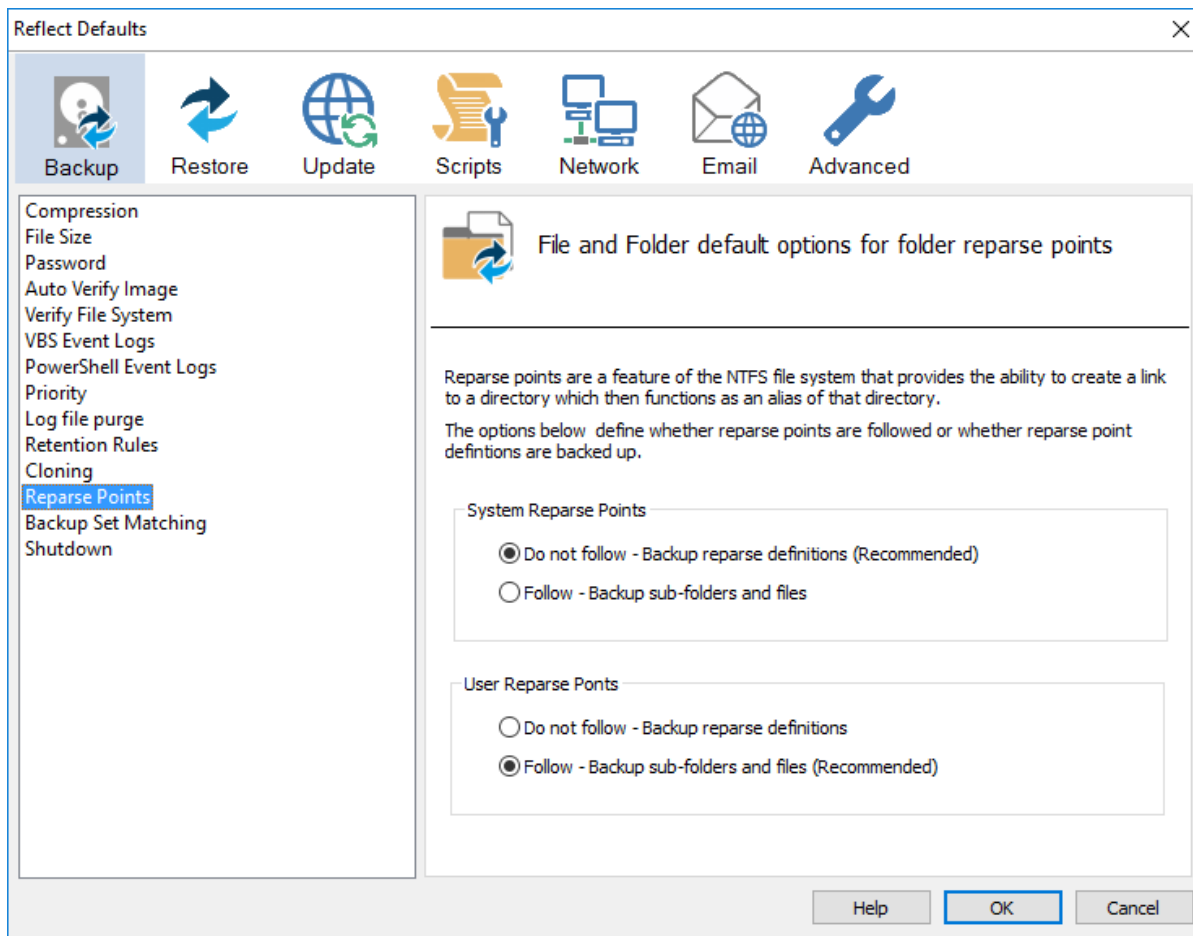
- Reparse points are a feature of the NTFS file system that provides the ability to create a link to a directories which then fictions as an alias of that directory.

e.g. Reparse point is the folder "Documents and Settings" which when followed points (or expands) to a number of other folders. If followed then all folders the reparse point "contents" will be included in the backup.

The options below define whether reparse points are followed or whether reparse point definitions are backed up:

Option	Description	
System Reparse Points	**Do not follow**	Only backup the **Reparse Definitions** (Recommended)
	Follow	Backup all the **Reparse Points**
User Reparse Points	**Do not follow**	Backup the **Reparse Definitions**
	Follow	Backup all the **Reparse Points** (Recommended)

Backup Set Matching options sets rules for retention that is going to be applied to your backup sets.
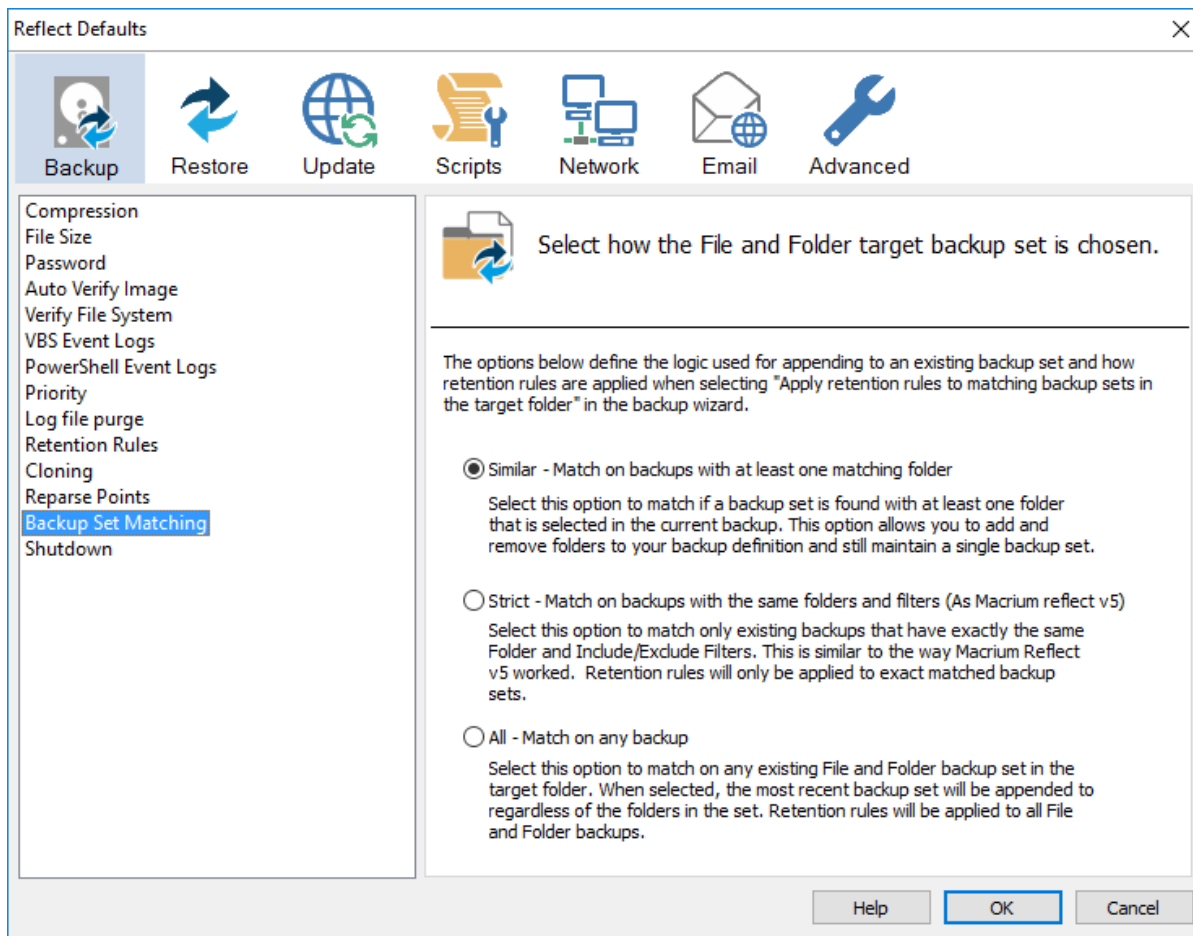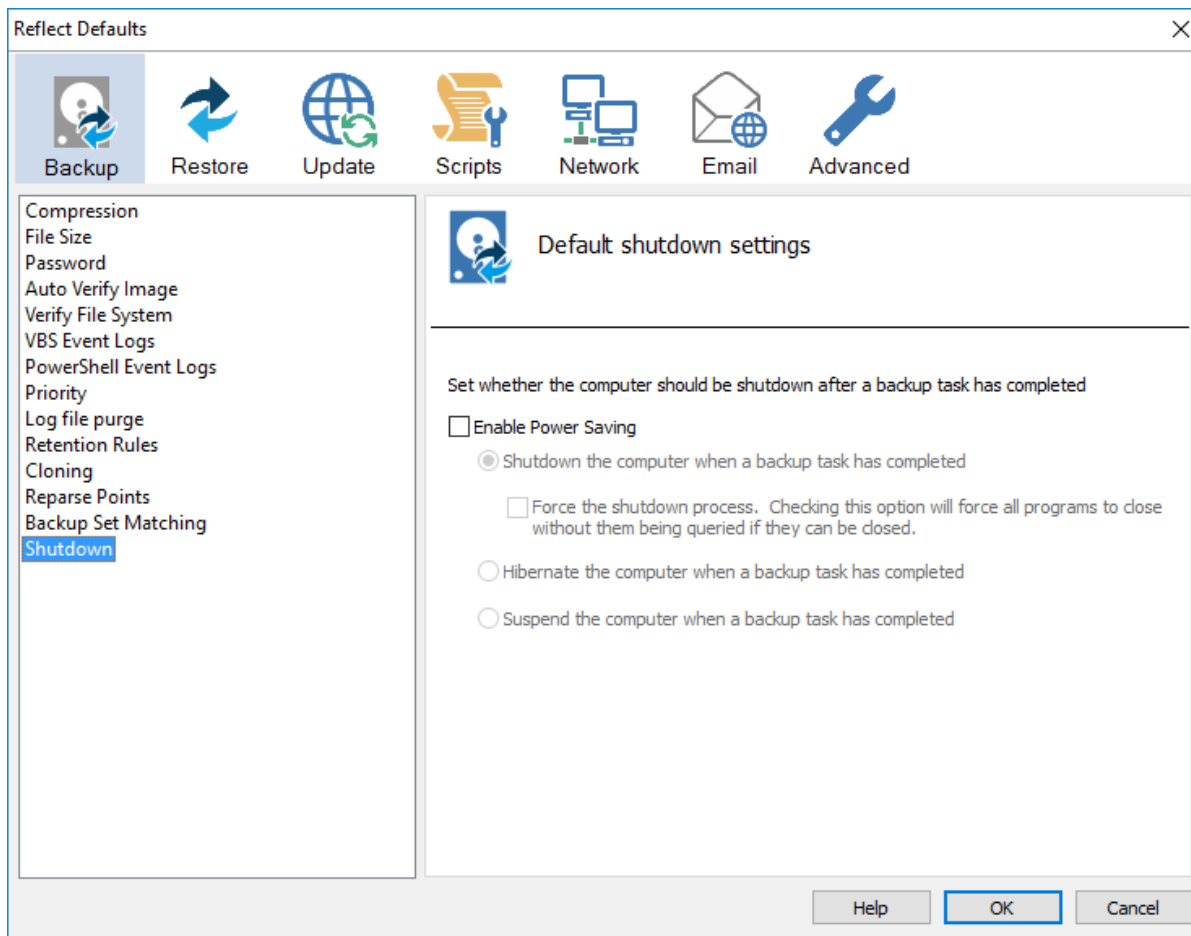
- The options below define the logic used for appending to an existing backup set and how retention rules are applied when selecting 'Apply retention rules to matching backup sets in the target folder' in the backup wizard.

Option	Description
Similar - Match on backups with at least one matching folder	Add and remove folders in your backup definition and still maintain a single backup set.
Strict - Match on backups with the same folders and filters **(As Macrium Reflect v5)**	Retention rules will only be applied to exact matched backup sets.
All - Matching on any backup	Retention rules will be applied to all File and Folder backups.

Shutdown sets whether the computer should be shutdown after a backup task has completed.

These options set the defaults for a new backup.

Reflect Defaults ✕

| Backup | Restore | Update | Scripts | Network | Email | Advanced |

Compression
File Size
Password
Auto Verify Image
Verify File System
VBS Event Logs
PowerShell Event Logs
Priority
Log file purge
Retention Rules
Cloning
Reparse Points
Backup Set Matching
Shutdown

Default shutdown settings

Set whether the computer should be shutdown after a backup task has completed

☐ Enable Power Saving

◉ Shutdown the computer when a backup task has completed

☐ Force the shutdown process. Checking this option will force all programs to close without them being queried if they can be closed.

○ Hibernate the computer when a backup task has completed

○ Suspend the computer when a backup task has completed

Help | OK | Cancel

Option	Description
Shutdown	This will **Shutdown** your computer after the backup is complete. A sub-option can be enabled to **Force the shutdown process** - All programs will be forced to close without being queried.
Hibernate	This will **Hibernate** your computer after the backup is complete
Suspend	This will put your computer to **Sleep** after the backup is complete.

Default Restore settings

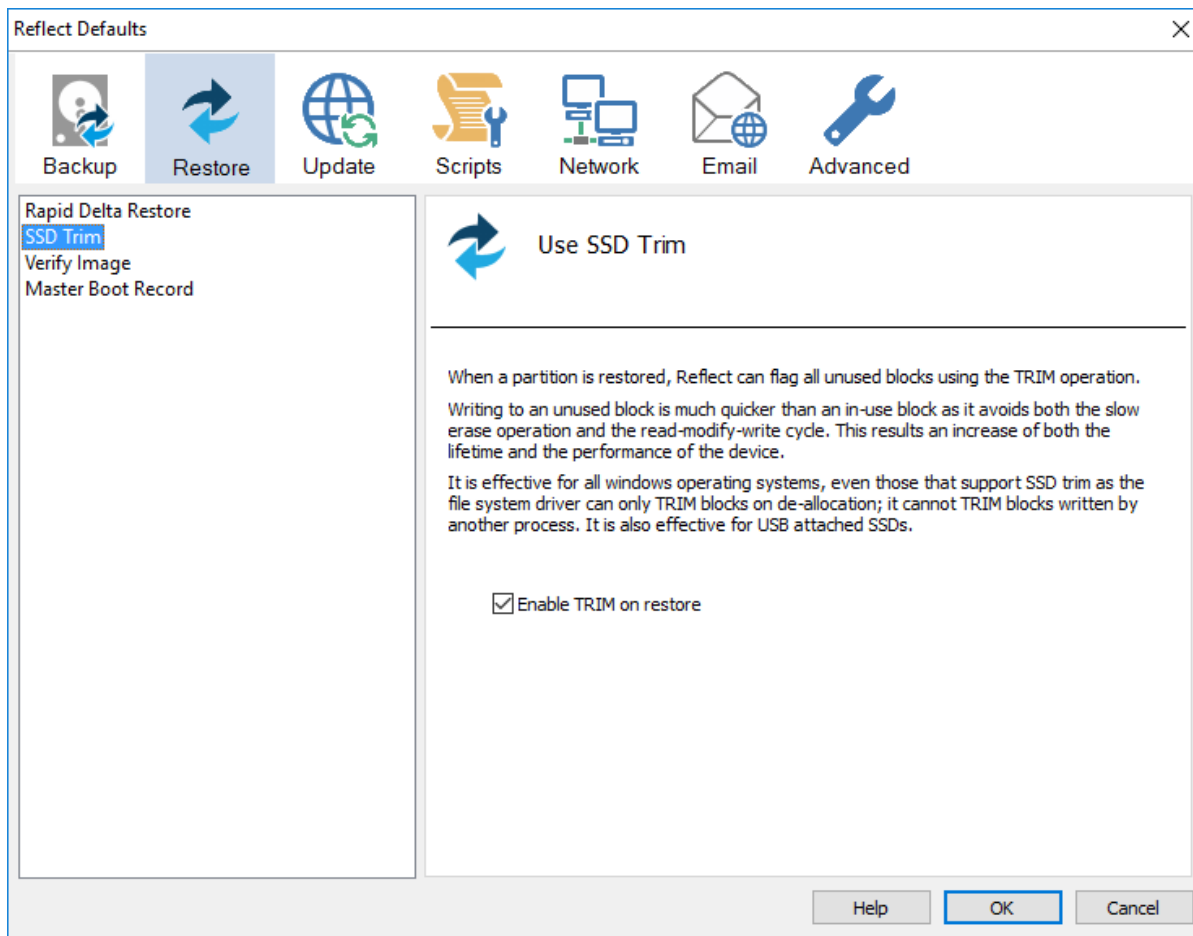**Rapid Delta Restore** will only restore the changed data blocks and not the whole backup file.

These options set the defaults for a new backup.
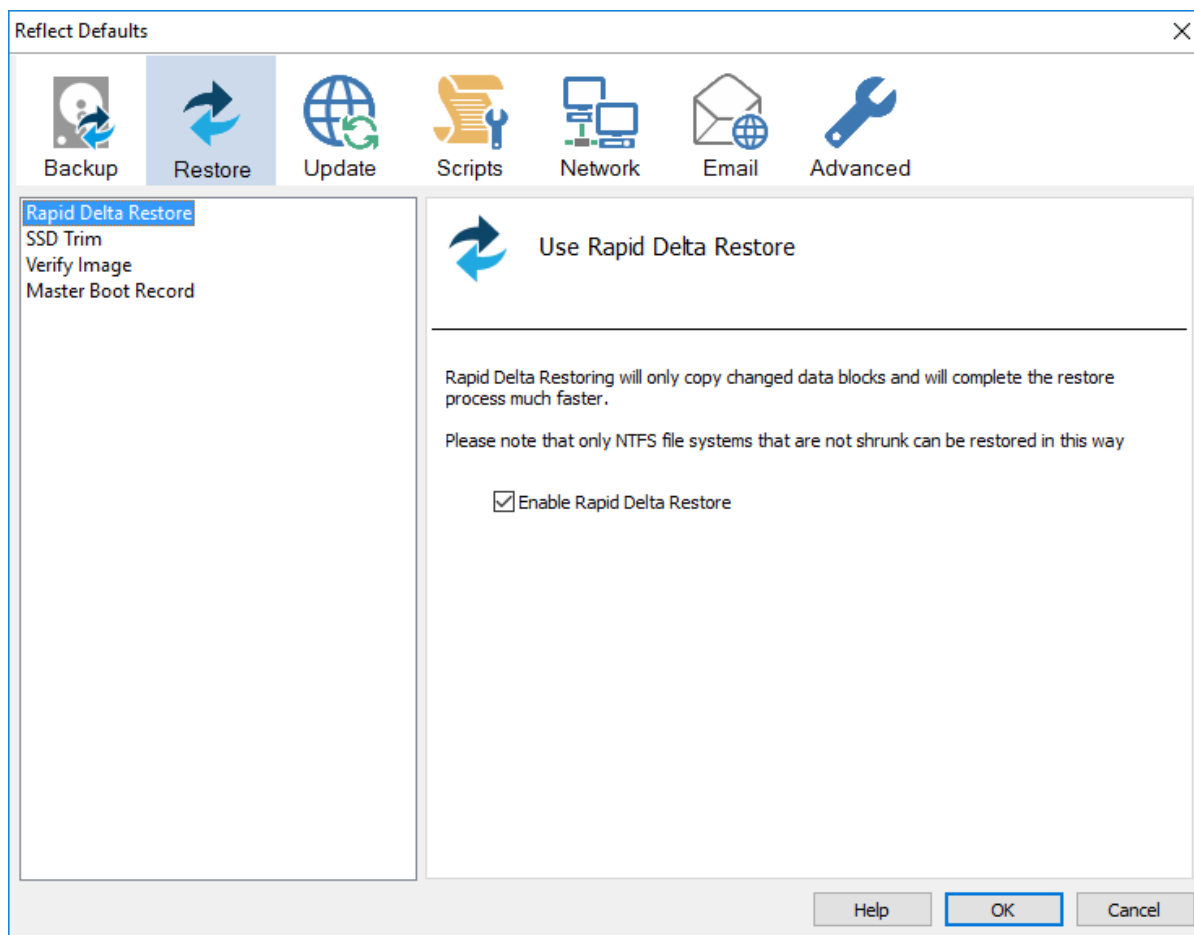
Enabling Rapid Delta Restore decreases the time it will take to restore a backup. Only NTFS file systems that are not shrunk can be restored this way. To find out more about RDR please click here.
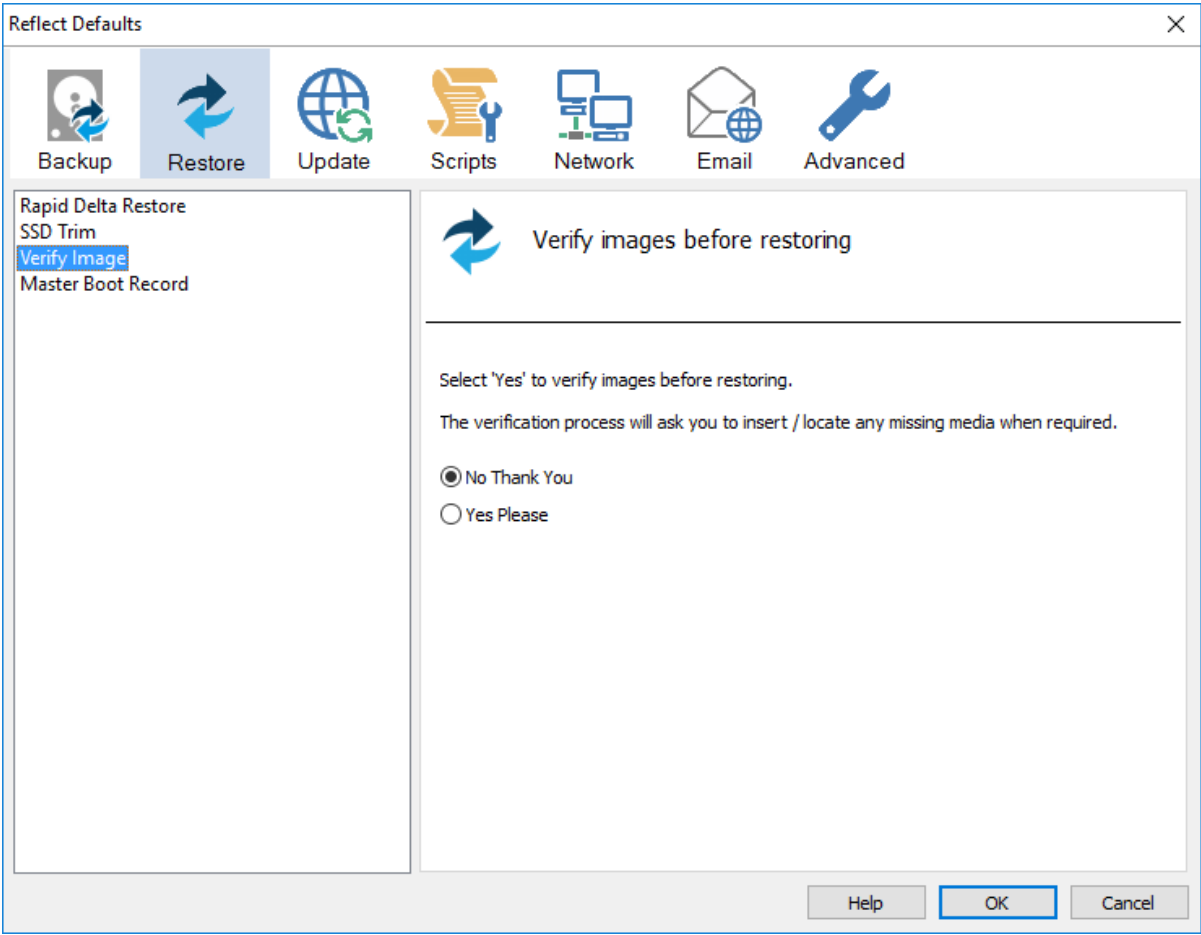
SSD Trim option enables SSD TRIM on restore.

These options set the defaults for a restore.

Reflect Defaults   ✕

| Backup | Restore | Update | Scripts | Network | Email | Advanced |

Rapid Delta Restore
SSD Trim
Verify Image
Master Boot Record

Use SSD Trim

When a partition is restored, Reflect can flag all unused blocks using the TRIM operation.

Writing to an unused block is much quicker than an in-use block as it avoids both the slow erase operation and the read-modify-write cycle. This results an increase of both the lifetime and the performance of the device.

It is effective for all windows operating systems, even those that support SSD trim as the file system driver can only TRIM blocks on de-allocation; it cannot TRIM blocks written by another process. It is also effective for USB attached SSDs.

☑ Enable TRIM on restore

| Help | OK | Cancel |

- When a partition is restored, Reflect can flag all unused blocks using the TRIM operation.
  Writing to an unused block is much quicker than an in-use block as it avoids both the slow erase operation and the read-modify-write cycle. This results an increase of both the lifetime and the performance of the device.
  It is effective for all Windows operating systems, even those that support SSD trim as the file system driver can only TRIM blocks on de-allocation; it cannot TRIM blocks written by another process. It is also effective for USB attached SSDs.

Verify Image option verifies images before restoring.

These options set the defaults for a restore.

Reflect Defaults

Backup	Restore	Update	Scripts	Network	Email	Advanced

Rapid Delta Restore
SSD Trim
Verify Image
Master Boot Record

Verify images before restoring

Select 'Yes' to verify images before restoring.

The verification process will ask you to insert / locate any missing media when required.

○ No Thank You
○ Yes Please

Help   OK   Cancel

Option	Description
No Thank You	Reflect will not verify your images before the Restore process.
Yes Please	Reflect will verify your images before the Restore process.

The verification process will ask you to insert / locate any missing media when required.

Master Boot Record options select whether to recover and replace the Master boot record from the backup.

These options set the defaults for a restore.
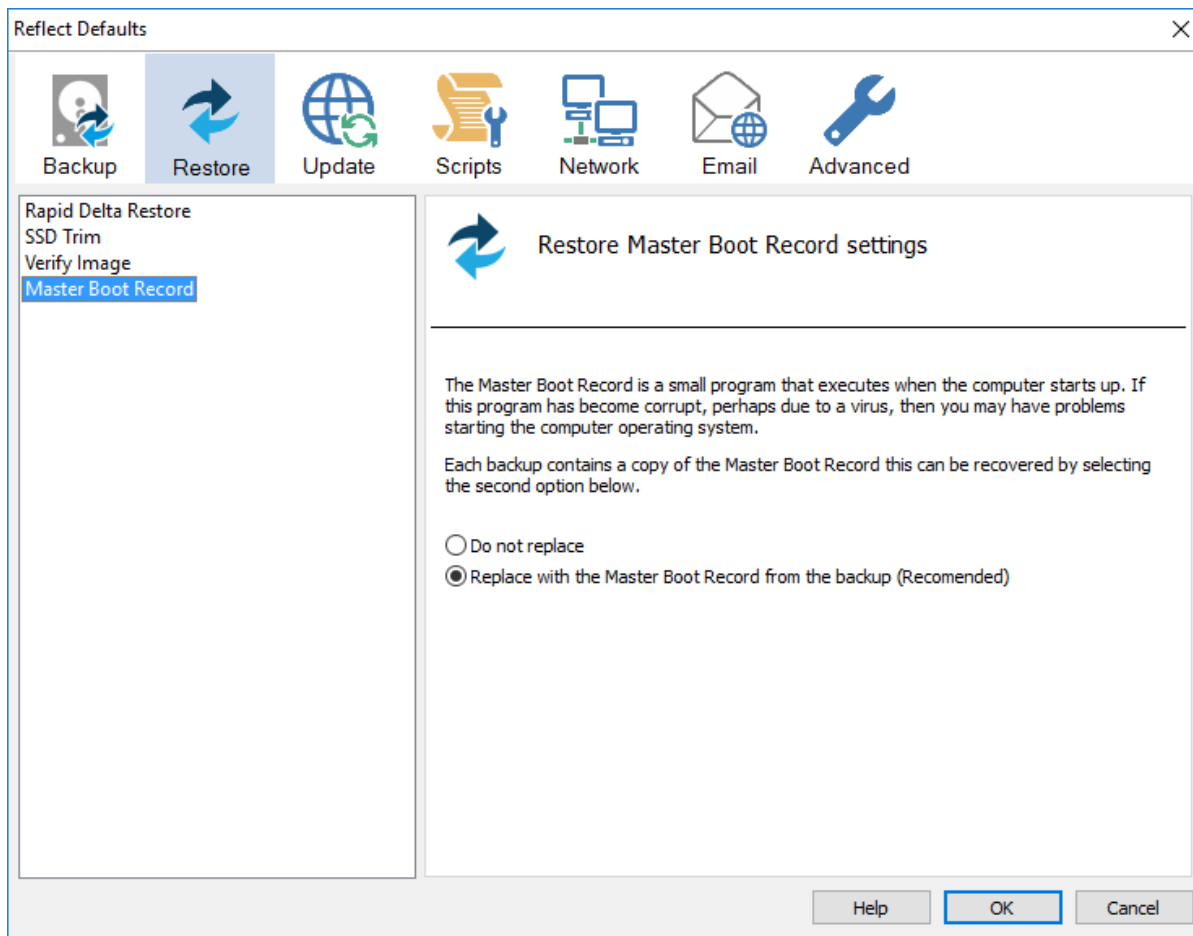
- Master Boot Record is a small program that executes when the computer starts up. If this program has become corrupt, perhaps due to a virus, then you may have problems starting the computers operating system.

Each backup contains a copy of the Master Boot Record that can be recovered by selecting the second option below:

Option	Description
Do not replace	Original MBR will be kept without replacing it with the MBR contained in the backup image.
Replace	MBR is replaced with the MBR from the the backup image.

Update Settings

Update options establish rules for updating Reflect.
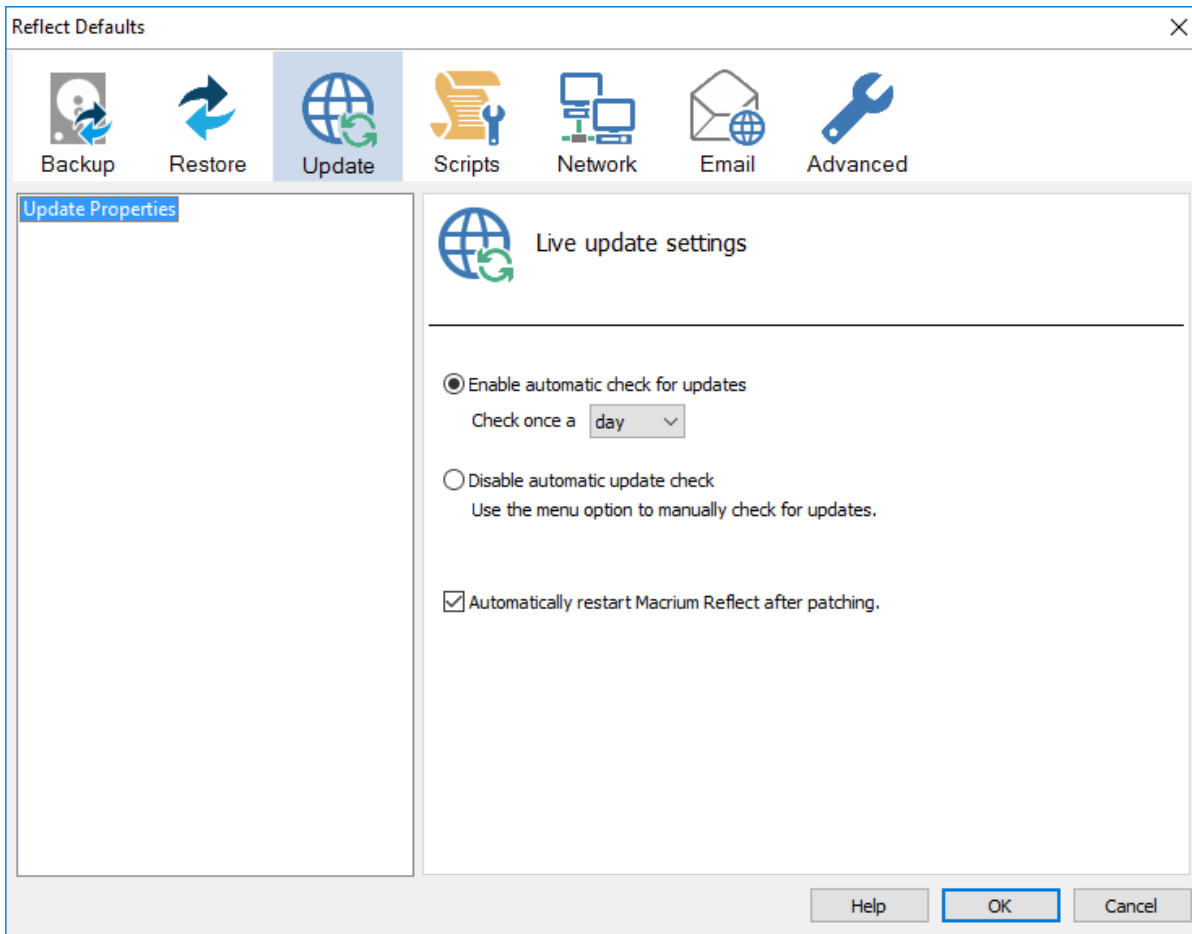
Option	Description
Enable automatic check for updates	Chose when Reflect checks for updates.
Disable automatic update check	Reflect will not check for updates automatically; updates will need to be manually checked for by using the update button.
Automatically restart Reflect after patching	Reflect will automatically restart once the update has been downloaded and installed.

Scripts

VBScript Logging options enable file logging in the same directory as your script file.

These options set the defaults for a script.
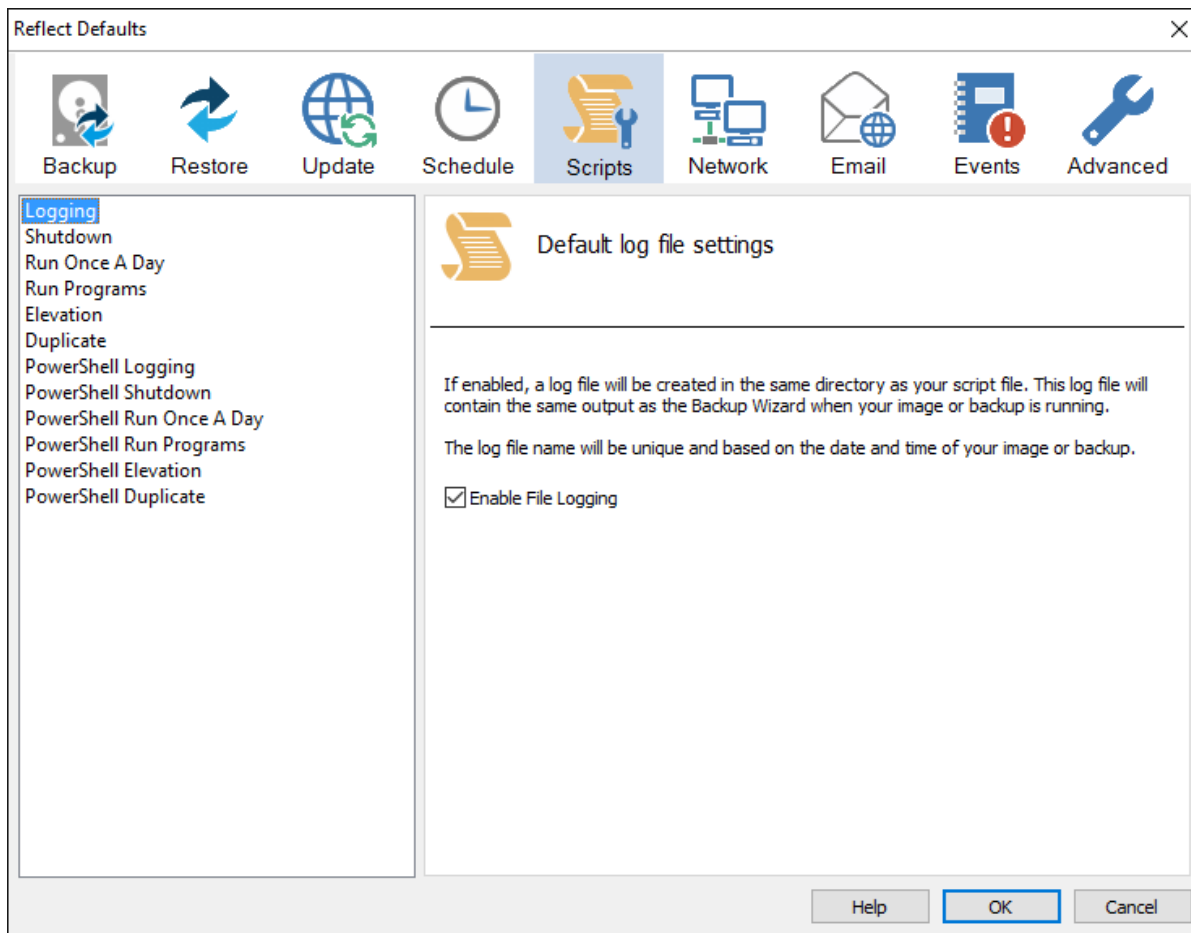
- If enabled, the log file created will contain the same output as the Backup Wizard when your image or backup is running. The log file name will be left unique and based on the date and time of your image or backup.

VBScript Shutdown option sets whether the PC should shutdown or log off at the end of the script.

These options set the defaults for a script.
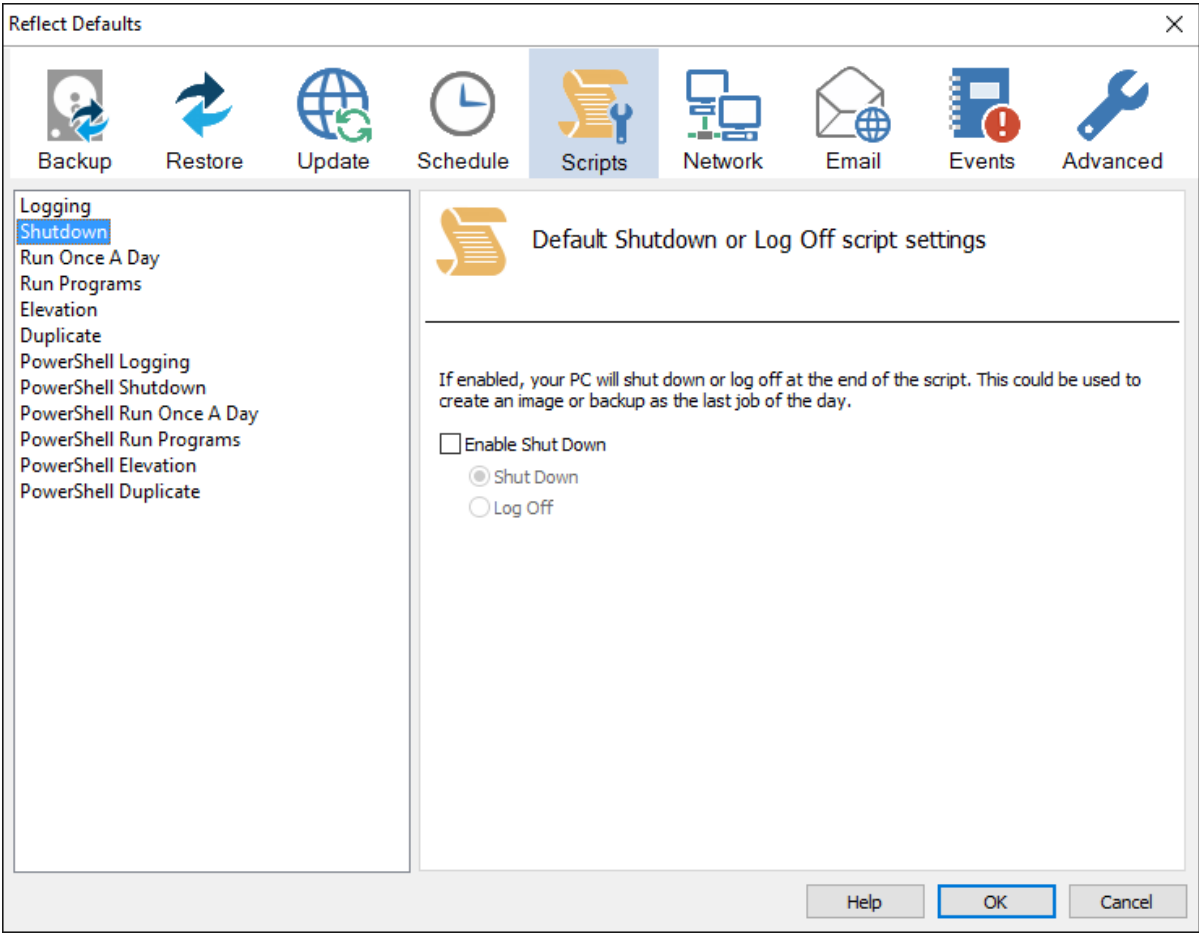
This could be used when creating a backup as the last job of the day.

```
' Issue shutdown command

    objShell.Run "shutdown -s"
```

Option	Description
Shut Down	☑ Enable Shut Down ◉ Shut Down ○ Log Off Your PC will shutdown after the script has been ran.
Log Off	☑ Enable Shut Down ○ Shut Down ◉ Log Off Your PC will Log Off after the script has been ran.

VBScript Run Once A Day option executes a script the first time you logon, logoff or shutdown.

These options set the defaults for a script.

Reflect Defaults   ✕

| Backup | Restore | Update | Schedule | **Scripts** | Network | Email | Events | Advanced |

Logging
Shutdown
Run Once A Day
Run Programs
Elevation
Duplicate
PowerShell Logging
PowerShell Shutdown
PowerShell Run Once A Day
PowerShell Run Programs
PowerShell Elevation
PowerShell Duplicate

Default Once A Day settings

It can be useful to execute a script the first time you logon, logoff or shut down. Using this component you can be sure that an event triggered task only runs this script once a day.

☐ Run Once a Day.

[Help]   [OK]   [Cancel]

```
' The following function call ensures that this script only runs once a day
    If HasRunToday Then
        WScript.Quit
    End If

...........

'*****************************************************************************
'* Function: HasRunToday
'*
'* Purpose:  determines if this script has run today
'*
'*
'* Input:    None
'* Output:   true if has run today false otherwise
'*
'*****************************************************************************
Function HasRunToday
    Dim RegScriptKey
    Dim LastRunDate
    Dim objShell

    Set objShell = WScript.CreateObject("WScript.Shell")
    RegScriptKey = "HKCU\SOFTWARE\Macrium\Reflect\Scripts\" & WScript.ScriptFullName & "\LastRun"
    'Check if script has run today
    ON ERROR RESUME NEXT
    LastRunDate = objShell.RegRead(RegScriptKey)
    If LastRunDate =  cstr(Date) Then
         HasRunToday = true
    Else
         objShell.RegWrite RegScriptKey, Date,"REG_SZ"
         HasRunToday  = false
    End If
    Set objShell = nothing
End Function
```

Option	Description
Run once a day	☑ Run Once a Day. The script will run only once a day.

This is useful if you want a backup to happen at first login or shutdown.

VBScript Run Programs selects programs or scripts to run at the start or end of the script.

These options set the defaults for a script.
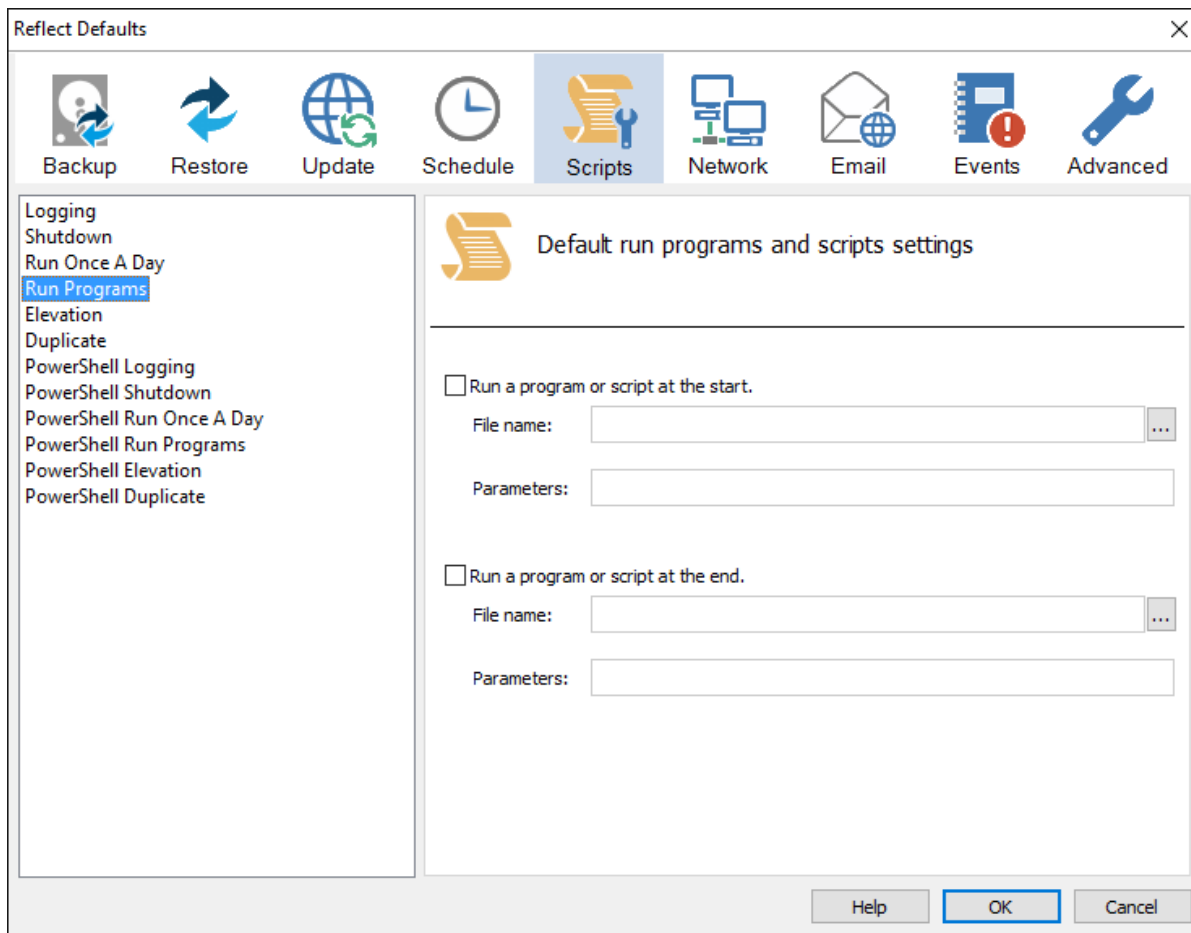
```
' Run program before backup
    objShell.Run """C:\Users\Public\Admin\Before backup.bat""", 1, true
' Do the backup
    ExitCode = Backup ("""C:\Program Files\Macrium\Reflect\reflect.exe"" -e -w <BACKUP_TYPE> ""C:
\Users\Dev\Documents\Reflect\example.xml""")
' Run program after backup
    objShell.Run """C:\users\public\Admin\After backup.exe""", 1, true
```

Option	Description
File name	Write the path and executable file name that you wish to run at the start of the script.
Parameters	Include optional command line parameters for the program.
File name	Write the path and executable file name that you wish to run at the end of the script.
Parameters	Include optional command line parameters for the program.

VBScript Elevation is used for Vista and later, to run all programs with elevated privileges.

These options set the defaults for a script.

Reflect Defaults   ×

| Backup | Restore | Update | Schedule | Scripts | Network | Email | Events | Advanced |

Logging
Shutdown
Run Once A Day
Run Programs
Elevation
Duplicate
PowerShell Logging
PowerShell Shutdown
PowerShell Run Once A Day
PowerShell Run Programs
PowerShell Elevation
PowerShell Duplicate

Default VB Script elevation settings

When enabled scripts will prompt for elevation and run all called programs with elevated privileges.

Note: The return code from these scripts will always be zero. If you are running these scripts from the scheduler then this option is not necessary. The scheduler will elevate the script at start-up.

☑ Enable elevation. (Vista/Windows 7 Only)

| Help | OK | Cancel |

```
Sub VBMain()
    Dim objShell
    Dim ExitCode
' Elevate this script for Admin privileges in Vista
    Elevate

.................


'******************************************************************************
'* Sub:     Elevate
'*
'* Purpose: Elevates this script for Vista UAC.
'*          This means that only one UAC Elevation prompt is displayed and
'*          functions/programs will not fail if they require admin privileges
'*
'* Input:   None
'* Output:  None
'*
'******************************************************************************
Sub Elevate
    Dim ShellApp
    Dim objShell
    Dim objWshProcessEnv
    ' Don't elevate if run from Windows Task Scheduler
    If WScript.Arguments.Count > 0 Then
        If WScript.Arguments.Item(0) = "-s" then
            Exit Sub
        End If
    End If
    Set objShell = WScript.CreateObject("WScript.Shell")
    Set objWshProcessEnv = objShell.Environment("PROCESS")
    If objWshProcessEnv("ELEVATED_APP") <> "True" Then
        objWshProcessEnv("ELEVATED_APP") = "True"
        Set ShellApp = CreateObject("Shell.Application")
        Call ShellApp.ShellExecute("""" &  WScript.FullName  & """", """" & WScript.ScriptFullName & """" & " "
& GetBackupTypeParameter,  , "runas")
        set ShellApp = nothing
        Set objWshProcessEnv = nothing
        wscript.quit
    End If
    Set objWshProcessEnv = nothing
    Set objShell = nothing
End Sub
```

The return code from these scripts will always be zero. If you are running these scripts from the scheduler then this option is not necessary. The scheduler will elevate the script at startup.

Option	Description
Enable Elevation	Enables UAC elevation for the entire script.
	This enables functions and programs to run outside the context of Macrium Reflect without requesting further elevation.

VBScript Duplicate option creates a copy of the backup to a chosen directory when the backup completes

These options set the defaults for a script.
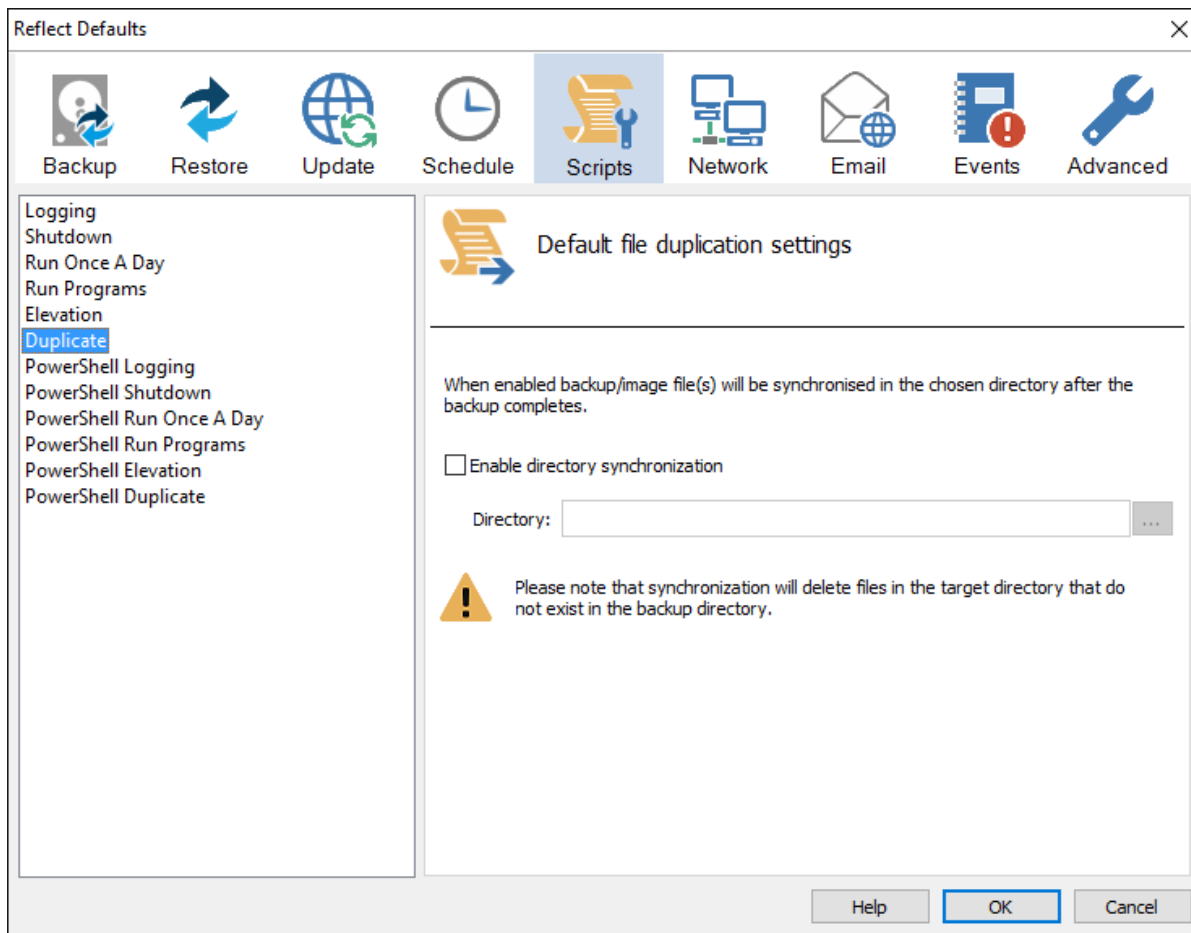
```
    SynchroniseDirectories ExitCode, "C:\C++"
.................


'*****************************************************************************
'* Function: RobocopyExists
'*
'* Purpose:  determines whether the program Robocopy can be called from the
'*           command line.
'*
'* Input:    None
'* Output:   boolean - true if Robocopy exists
'*
'*****************************************************************************
Function RobocopyExists()
    On Error Resume Next
    Dim objShell
    Dim objExec
    Set objShell = WScript.CreateObject("WScript.Shell")
    Set objExec = objShell.Exec ("robocopy.exe", 0)

    If Err.Number = 0 Then
        RobocopyExists = True
    Else
        RobocopyExists = False
    End If
End Function


'*****************************************************************************
'* Function: SynchroniseDirectories
```

```
'*
'* Purpose:  Copies all files created by the previous backup / image
'*           to a supplied directory. Uses Macrium environment variables to
'*           determine which files to copy.
'*
'* Input:    ExitCode - The exit code of the last backup
'*           strBackupDirectory - Directory to copy to
'*
'******************************************************************************
Function SynchroniseDirectories(Byval ExitCode, Byval strSyncDirectory)
    Dim objShell
    Dim objWshProcessEnv
    Dim strEnvPrefix
    Dim strBackupDirectory
    Dim strCmdLine
    Dim iReturnCode
    Dim fs
    Dim objSyncFiles
    Dim objBackupDirectory
    Dim objSyncDirectory
    Dim objBackupFile
    Dim objSyncFile
    Dim strExtension
    Dim dateBackupFile
    Dim dateSyncFile

' Only copy files if backup was successful
    if ExitCode <> 0 Then
        Exit Function
    End If

    Set objShell         = WScript.CreateObject("WScript.Shell")
    Set objWshProcessEnv = objShell.Environment("VOLATILE")

' Get the prefix for the last backup set
    strEnvPrefix = objWshProcessEnv("MACRIUM_PREFIX")

' Get the directory where we just created a backup
    strBackupDirectory = objWshProcessEnv(strEnvPrefix + "_DIRECTORY")

    If right(strBackupDirectory, 1) = "\" Then strBackupDirectory = left(strBackupDirectory, len
(strBackupDirectory)-1)
    If right(strSyncDirectory, 1)   = "\" Then strSyncDirectory   = left(strSyncDirectory, len(strSyncDirectory)
-1)

    If RobocopyExists Then
        strCmdLine = "robocopy ""<SOURCE>"" ""<DESTINATION>"" *.mr* /copy:DAT /lev:0 /purge /r:0"
    Else
    ' Robocopy does not exist - using xcopy
        Set fs = CreateObject("Scripting.FilesystemObject")
        Set objBackupDirectory = fs.GetFolder(strBackupDirectory)
        Set objSyncDirectory = fs.GetFolder(strSyncDirectory)

        For Each objSyncFile in objSyncDirectory.Files
            strExtension = fs.GetExtensionName(objSyncFile)
            dateSyncFile = objSyncFile.DateLastModified
            If Left(strExtension,2) = "mr" Then
            ' Check if file has been deleted
                If Not(fs.FileExists(strBackupDirectory+"\"+objSyncFile.name)) Then
                    fs.DeleteFile(objSyncFile)
                Else
                ' Check if file has been modified
                    Set objBackupFile = fs.GetFile(strBackupDirectory+"\"+objSyncFile.name)
                    dateBackupFile = objBackupFile.DateLastModified
                    If DateDiff("m", dateBackupFile, dateSyncFile) <> 0 Then
                        fs.DeleteFile(objSyncFile)
                    End If
                End If
            End If
        Next
```

```
        strCmdLine = "xcopy ""<SOURCE>\*.mr*"" ""<DESTINATION>"" /c /d /h /i /v /y"
    End If

    strCmdLine = Replace(strCmdLine,"<SOURCE>", strBackupDirectory)
    strCmdLine = Replace(strCmdLine,"<DESTINATION>", strSyncDirectory)
    iReturnCode = objShell.Run(strCmdLine,0,true)

    if iReturnCode <> 0 then
' Handle synchronisation error
    else
' Everything OK
    end if

' Clean up
    Set objShell       = nothing
    Set objWshProcessEnv = nothing
End Function
```

Option	Description
Enable file copy	If enabled the backup will create a copy of the files from the backup to the supplied destination.
Directory	Here you will need to enter the folder that you want to copy/archive your backup files to.

Please note that the synchronization will delete files in the target directory that do not exist in the backup directory.

Powershell Logging options enable file logging in the same directory as your script file.

These options set the defaults for a script.
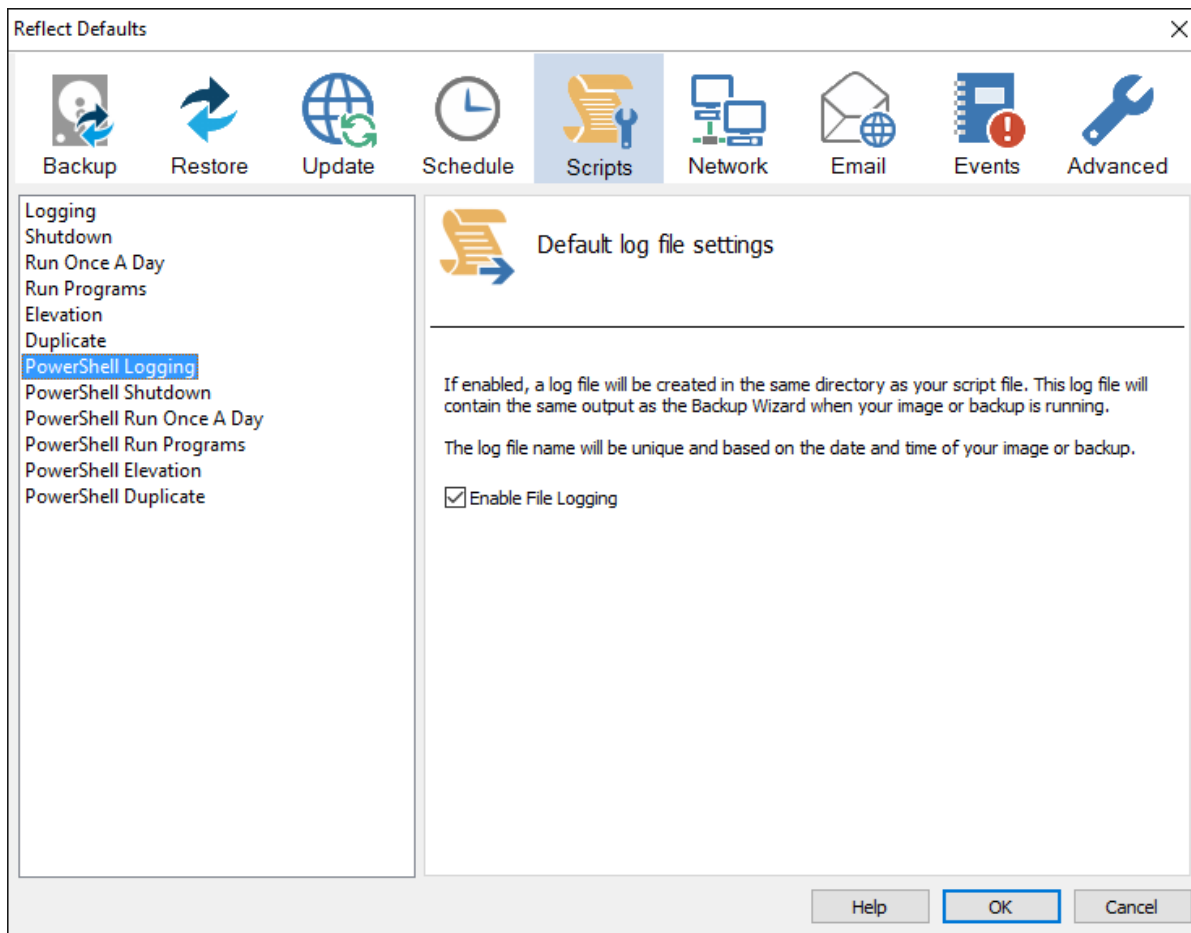
- If enabled, the log file created will contain the same output as the Backup Wizard when your image or backup is running. The log file name will be left unique and based on the date and time of your image or backup.

Powershell Shutdown option sets whether the PC should shutdown or log off at the end of the script.

These options set the defaults for a script.
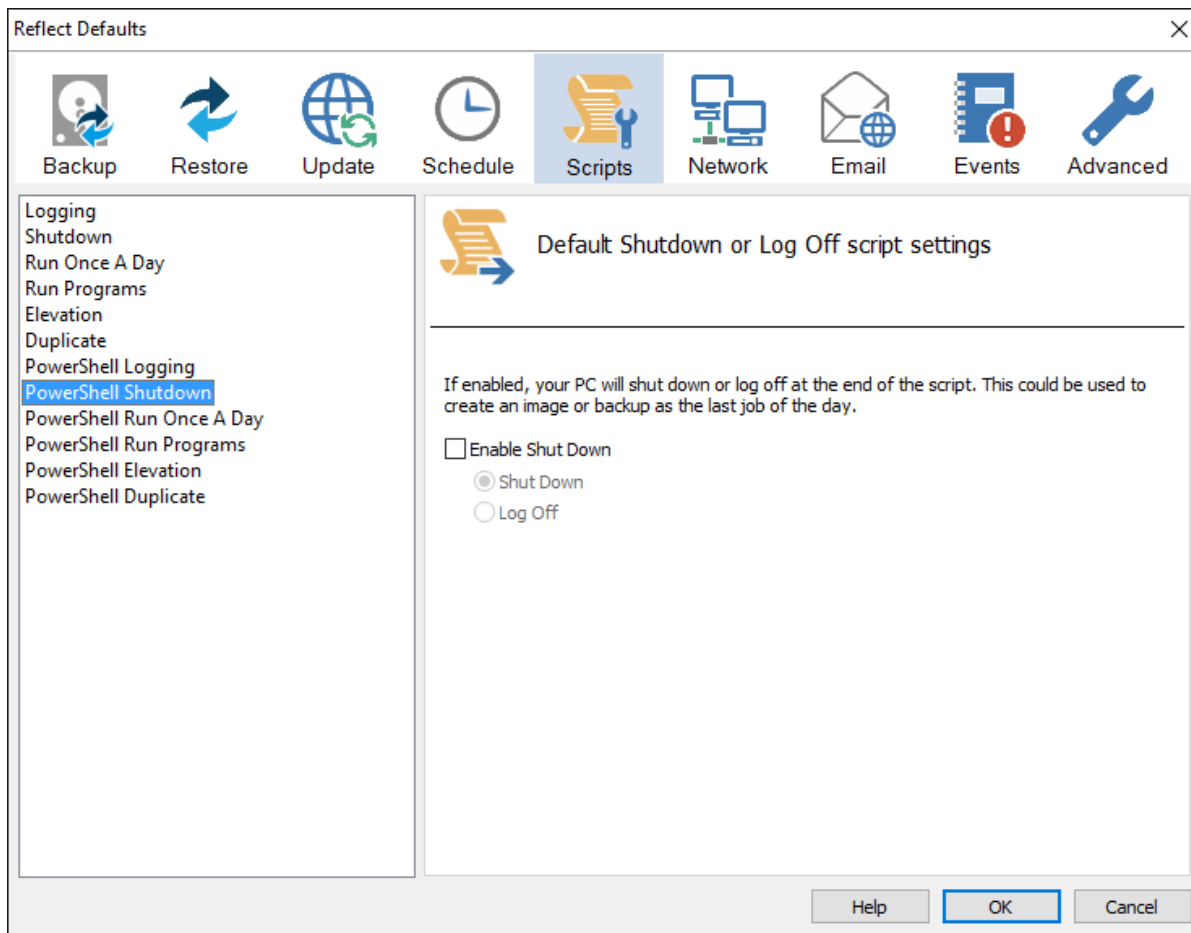
```
Write-Host ' * Initiating shutdown... ' -NoNewLine;
(Get-WMIObject Win32_OperatingSystem -ComputerName '.' -EnableAllPrivileges).Win32Shutdown(8);
```

Option	Description
Shut Down	 Your PC will shutdown after the script has been ran.
Log Off	 Your PC will Log Off after the script has been ran.

Powershell Run Once A Day option executes a script the first time you logon, logoff or shutdown.

These options set the defaults for a script.

Reflect Defaults   ✕

Backup	Restore	Update	Schedule	Scripts	Network	Email	Events	Advanced

Logging
Shutdown
Run Once A Day
Run Programs
Elevation
Duplicate
PowerShell Logging
PowerShell Shutdown
PowerShell Run Once A Day
PowerShell Run Programs
PowerShell Elevation
PowerShell Duplicate

Default Once A Day settings

It can be useful to execute a script the first time you logon, logoff or shut down. Using this component you can be sure that an event triggered task only runs this script once a day.

☐ Run Once a Day.

Help   OK   Cancel

```
  if (HasRunToday)
  {
    Write-Host ' * Script already executed today. Exiting...';
    Write-Host 'Script finished with exit code 0.';
    Exit 0;
  }

.................

#******************************************************************************
#* Func: HasRunToday
#*
#* Desc: determines if this script has run today
#*
#******************************************************************************
function HasRunToday()
{
    Write-Host ' * Checking last run time... ' -NoNewLine;
    $strRegPath    = 'HKCU:\Software\Macrium\Reflect\Scripts';
    $boolRanToday  = $false;
    $strDateToday  = Get-Date -UFormat %Y%m%d;
    if (Test-Path $strRegPath)
    {
        try
        {
            $strLastRunDate = (Get-ItemProperty -Path $strRegPath -Name $strScriptPath -ErrorAction
Stop).$strScriptPath
            if ($strLastRunDate -eq $strDateToday)
            {
                $boolRanToday = $true;
            }
        } catch { };
    }
    if (!$boolRanToday)
    {
        Set-ItemProperty -Path $strRegPath -Name $strScriptPath -Value $strDateToday;
    }
    Write-Host 'Done.';
    return $boolRanToday;
}
```

Option	Description
Run once a day	☑ Run Once a Day. The script will run only once a day.

This is useful if you want a backup to happen at first login or shutdown.

Powershell Run Programs selects programs or scripts to run at the start or end of the script.

These options set the defaults for a script.

```
   Write-Host ' * Executing "at start" program... ' -NoNewLine;
   $strRunAtStartApp  = 'C:\Users\Public\Admin\Before backup.bat';
   $strRunAtStartArgs = '';
   if ([string]::IsNullOrEmpty($strRunAtStartArgs))
   {
     Start-Process -FilePath $strRunAtStartApp;
   }
   else
   {
     Start-Process -FilePath $strRunAtStartApp -ArgumentList $strRunAtStartArgs;
   }

......

   Write-Host ' * Executing "at end" program... ' -NoNewLine;
   $strRunAtEndApp  = 'C:\Users\Public\Admin\After backup.exe';
   $strRunAtEndArgs = '';
   if ([string]::IsNullOrEmpty($strRunAtEndArgs))
   {
     Start-Process -FilePath $strRunAtEndApp;
   }
   else
   {
     Start-Process -FilePath $strRunAtEndApp -ArgumentList $strRunAtEndArgs;
   }
```

Option	Description
File name	Write the path and executable file name that you wish to run at the start of the script.

Parameters	Include optional command line parameters for the program.
File name	Write the path and executable file name that you wish to run at the end of the script.
Parameters	Include optional command line parameters for the program.

Powershell Elevation is used for Vista and later, to run all programs with elevated privileges.

These options set the defaults for a script.
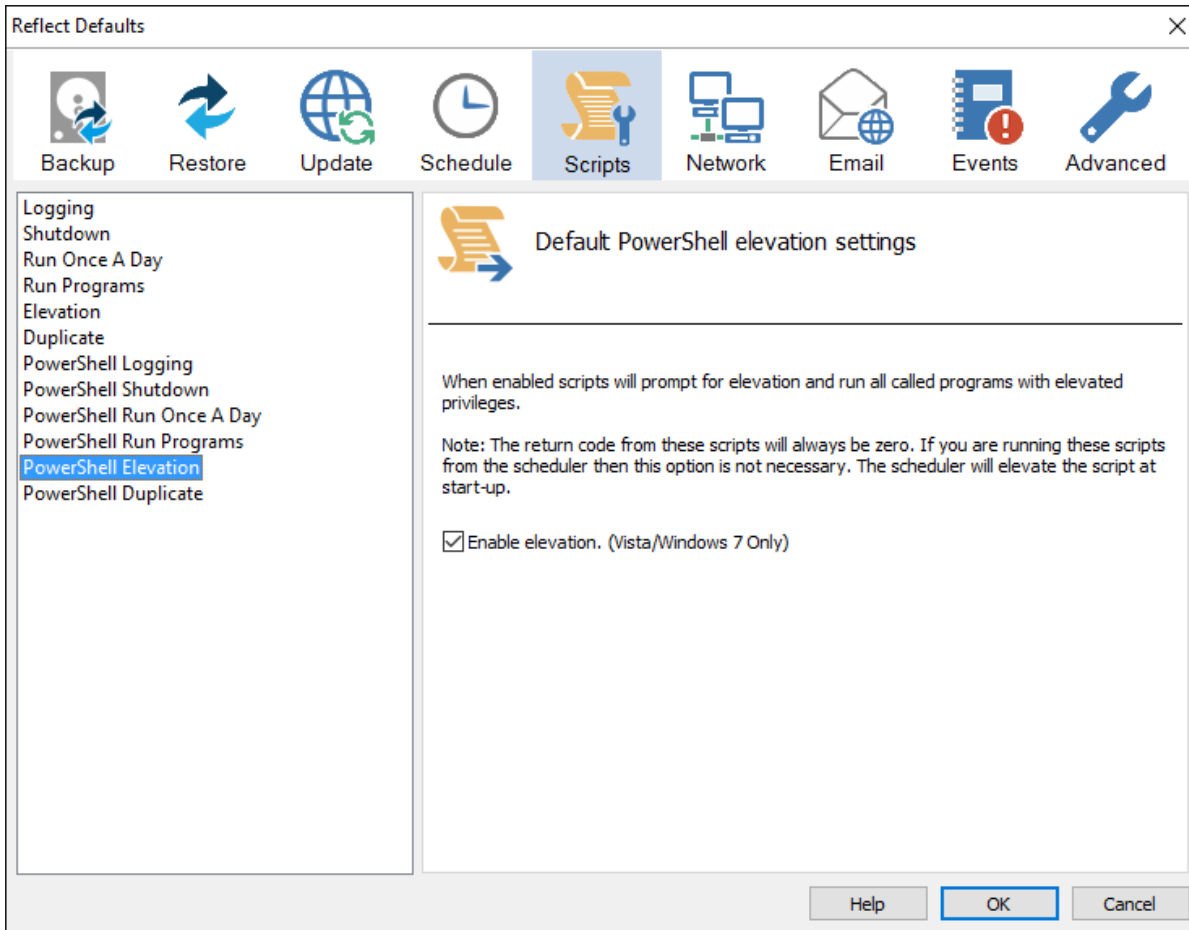
```
function Main()
{
  Write-Host 'PowerShell script for Macrium Reflect Backup Definition File';
  Write-Host "BDF: $strXmlFilePath";
  Elevate;

.................


#*******************************************************************************
#* Func: Elevate
#*
#* Desc: Elevates this script for UAC.
#*        This means that only one UAC Elevation prompt is displayed and
#*        functions/programs will not fail if they require admin privileges.
#*
#*******************************************************************************
function Elevate()
{
  # Only elevate if not ran from the task scheduler.
  Write-Host ' * Checking elevated access rights... ' -NoNewLine;
  if (-Not $s)
  {
    # Check to see if we are currently running "as Administrator"
    if (!([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole
([Security.Principal.WindowsBuiltInRole]"Administrator"))
    {
      $ElevatedProcess = new-object System.Diagnostics.ProcessStartInfo "PowerShell";
      # Specify the current script path and name as a parameter
      $strType = GetBackupTypeParameter;
      $ElevatedProcess.Arguments = "-ExecutionPolicy Bypass & '" + $script:MyInvocation.MyCommand.Path + "'
$strType";
      # Indicate that the process should be elevated
      $ElevatedProcess.Verb = "runas";
      # Start the new process
      [System.Diagnostics.Process]::Start($ElevatedProcess);
      # Exit this unelevated script with exit code for "Error: Not elevated"
      Exit 3;
    }
  }
  Write-Host 'Done.';
}
```

The return code from these scripts will always be zero. If you are running these scripts from the scheduler then this option is not necessary. The scheduler will elevate the script at startup.

Option	Description
Enable Elevation	Enables UAC elevation for the entire script. This enables functions and programs to run outside the context of Macrium Reflect without requesting further elevation.

Powershell Duplicate option creates a copy of the backup to a chosen directory when the backup completes.

These options set the defaults for a script.

Reflect Defaults ✕

| Backup | Restore | Update | Schedule | Scripts | Network | Email | Events | Advanced |

Logging
Shutdown
Run Once A Day
Run Programs
Elevation
Duplicate
PowerShell Logging
PowerShell Shutdown
PowerShell Run Once A Day
PowerShell Run Programs
PowerShell Elevation
PowerShell Duplicate

Default file duplication settings

When enabled backup/image file(s) will be synchronised in the chosen directory after the backup completes.

☐ Enable directory synchronization

Directory: [   ] [...]

⚠ Please note that synchronization will delete files in the target directory that do not exist in the backup directory.

[Help] [OK] [Cancel]

```
  if ($iExitCode -eq 0) # if backup completed successfully...
  {
    $strBackupDir = GetLastBackupPath;
    if (![string]::IsNullOrEmpty($strBackupDir))
    {
      SynchroniseDirectories $strBackupDir 'D:\Archive';
    }
  }

...............

#*******************************************************************************
#* Func: SynchroniseDirectories
#*
#* Desc: Copies all Macrium Reflect files to a supplied directory.
#*
#*******************************************************************************
function SynchroniseDirectories($strSrcDir, $strDstDir)
{
  Write-Host ' * Synchronising directories... ' -NoNewLine;
  if (Get-Command robocopy -ErrorAction SilentlyContinue)
  {
    # robocopy is available...
    # /copy:DAT - D:Data
    #             A:Attributes
    #             T:Time stamps
    # /purge    - deletes destination files and directories that no longer
    #               exist in the source
    # /lev:0    - Does not copy subdirectories
    &robocopy $strSrcDir $strDstDir *.mr* /copy:DAT /lev:0 /purge /r:0 | Out-Null
  }
  else
  {
    # Fall back to xcopy...

    # Delete files from the target directory not present in the source directory
    $strDstDirChildren = $strDstDir+"\*";
    Get-ChildItem $strDstDirChildren -include "*.mr*" | Foreach-Object {
      $strMaybeDeletedSrcFile = $strSrcDir + '\' + $_.Name;
      if (-not (Test-Path $strMaybeDeletedSrcFile))
      {
        Remove-Item $_
      }
      else
      {
        $SrcFileTime = [datetime](Get-ItemProperty -Path $strMaybeDeletedSrcFile -Name LastWriteTime).
lastwritetime;
        $DstFileTime = [datetime](Get-ItemProperty -Path $_.FullName         -Name LastWriteTime).
lastwritetime;
        $SrcFileTimeString = $SrcFileTime.ToString("yyyyMMddHHmmss")
        $DstFileTimeString = $DstFileTime.ToString("yyyyMMddHHmmss")
        if ($SrcFileTimeString -ne $DstFileTimeString)
        {
          Remove-Item $_
        }
      }
    }
    # /c - Continues copying even if errors occur.
    # /d - Date check; only copies if file does not exist or is older.
    # /h - Copies hidden and system files.
    # /i - If the destination does not exist, and you are copying more than one
    #        file, this switch assumes that the destination is a folder.
    # /v - Verifies each new file.
    # /y - Overwrites existing files without prompting.
    &xcopy $strSrcDir\*.mr* $strDstDir /c /d /h /i /v /y | Out-Null
  }
  Write-Host 'Done.';
}
```

Option	Description
Enable file copy	If enabled the backup will create a copy of the files from the backup to the supplied destination.
Directory	Here you will need to enter the folder that you want to copy/archive your backup files to.

Please note that the synchronization will delete files in the target directory that do not exist in the backup directory.

Network Settings

Windows contains a Network Credentials cache that enables network authentication details (User name and password) to be saved and re-used to logon automatically. However, scheduled tasks are run in Windows 'batch mode' and do not have access to saved credentials. Because of this you must enter network login details in the Macrium Reflect defaults to enable scheduled tasks to access and write to password protected network shares.

Option	Description
Add	Add network logon details for a new network location
Edit	Edit an existing location. **Note:** The **'Default'** location logon details will be used for all shares that aren't explicitly added.
Delete	Remove logon credentials

Add / Edit

Option	Description
\\Server\share	Enter the UNC path to the root of the network share. Do not enter any sub-folders in the path
Username	Enter the authenticated user that has access to the share
Password	Enter the users password.

You can also **select 'Default'** and **click 'Edit'** to provide default credentials to be used for all network shares:

Email Settings

- For many users, email notification of success or failure of a backup helps them keep on top of the status of their backups, particularly with scheduled backups which take place in the background.

- Select **'Email'** > **'Email Server**'.

Reflect Defaults
Backup
Restore
Update
Schedule
Scripts
Network
Email
Events
Advanced
Email Server
Email Success
Email Failure
Test
Help
OK
Cancel

Email Server Settings for Macrium Reflect

Senders Email: Enter the sender email address
Authentication: Auto Detect
SMTP Username: Enter server username
SMTP Password: Enter server password
SMTP Server: Enter host name
Connection Type: Plain Text (No Security)
SMTP Port: 25
Test Recipients: Enter test recipients and click Test, separate recipients with ;
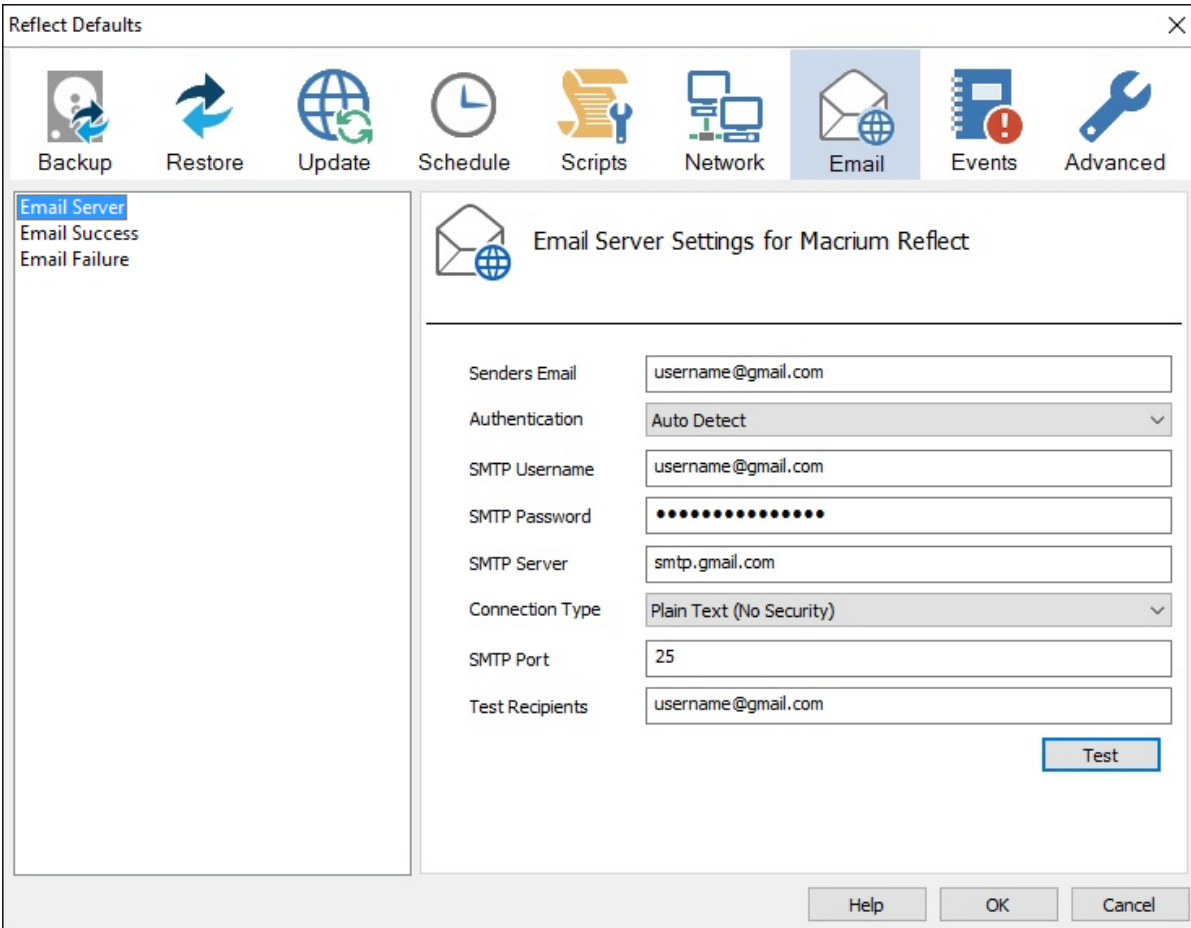
Email Server options description:

Option	Description
Senders Email	Your e-mail address associated with your ISP or Gmail account etc. Note: If you use e-mail software such as MS Outlook, you can find the settings under **Account Settings** for e-mail address, user-name and Server.
Authentication	Authentication options are set by your email provider/server. Auto Detect None Auto Detect Challenge/Response Authentication (CRAM-MD5) Secure Username/Password login (AUTH LOGIN) Username/Password login (AUTH PLAIN) Microsoft NT Lan Manager (NTLM) Note: If you do not know your authentication settings, try **Auto Detect**.
SMTP Username	The user name associated with your e-mail account. This is essentially your email address.
SMTP Password	This is the password for your e-mail Server.
SMTP Server	This is the outgoing / SMTP Server setting or IP address.

Connection Type	This is the setting for the way Macrium Reflect will contact the Server. Plain Text (No Security) Plain Text (No Security) Secure Sockets (SSL/TLS) Transport Layer Security (STARTTLS) Note: These settings can be found from your email provider.
SMTP Port	This is associated with **Connection Type** and is the port number that the SMTP server is listening on. Note: These settings can be found from your email provider.
Test Recipients	This option is used to test the configured settings; enter your e-mail address in that field and click **Test** to send a test message to your email.

The following examples show completed Server settings for a typical ISP.

- Click **OK** to finish.

Note: If you are using your Gmail account then please see here: Using Gmail SMTP Server for sending backup notification emails

Email success settings define who receives emails regarding the success of a backup and what message they receive.

- Select **Email Success**
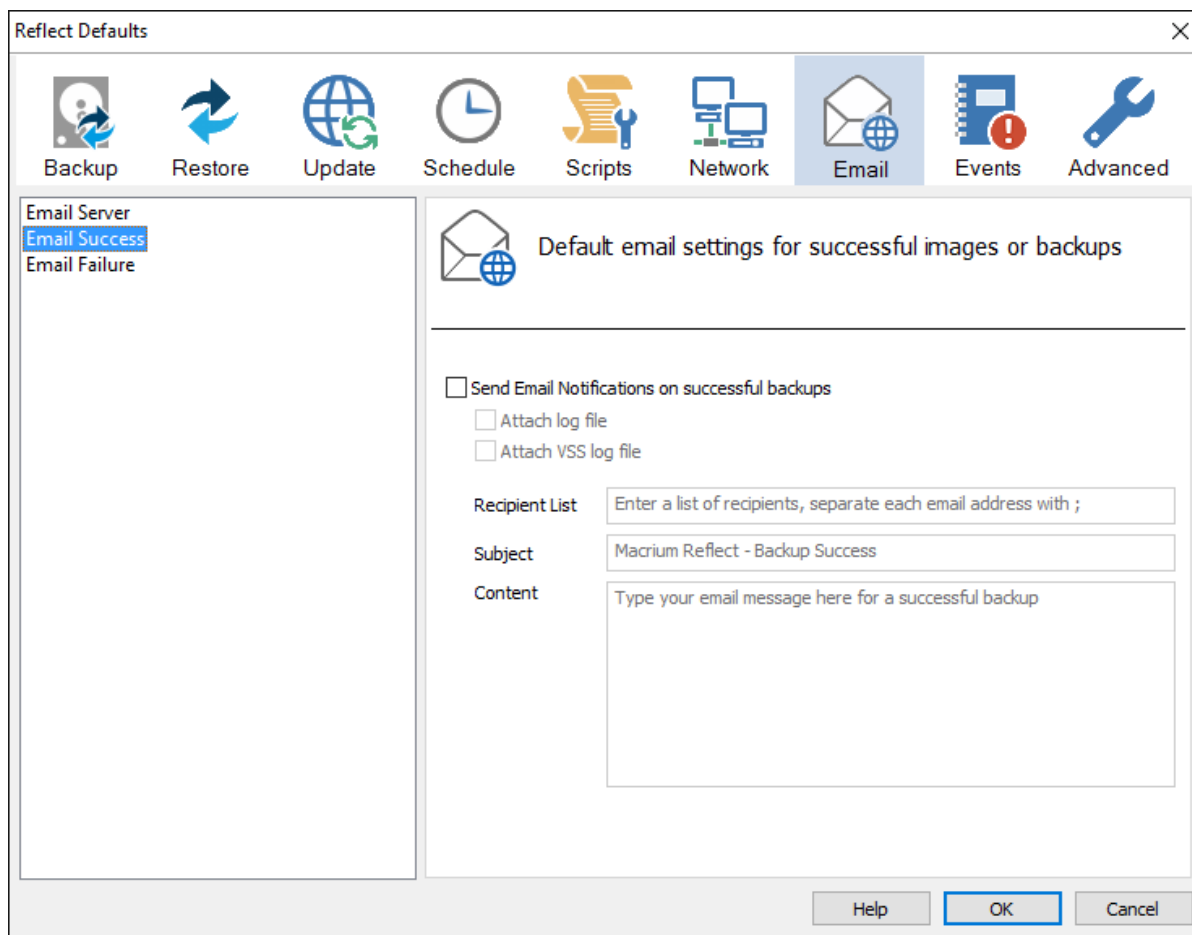
Reflect Defaults ✕

| Backup | Restore | Update | Schedule | Scripts | Network | Email | Events | Advanced |

Email Server
Email Success
Email Failure

Default email settings for successful images or backups

☐ Send Email Notifications on successful backups
   ☐ Attach log file
   ☐ Attach VSS log file

Recipient List   Enter a list of recipients, separate each email address with ;

Subject   Macrium Reflect - Backup Success

Content   Type your email message here for a successful backup

| Help | OK | Cancel |

- Select **Send Email Notifications on successful backups**.

Option	Description
Attach log file	Will attach the log file from the creation of the backup.
Attach VSS log file	Will attach the VSS log with the events made during the backup.

- Enter the email addresses of all recipients in **Recipient List**, separating each email address with a semi-colon.

- Enter the subject for the email in **Subject**.

- Enter a message to be sent regarding the email in **Content**, include the PC that generated the success is identified.
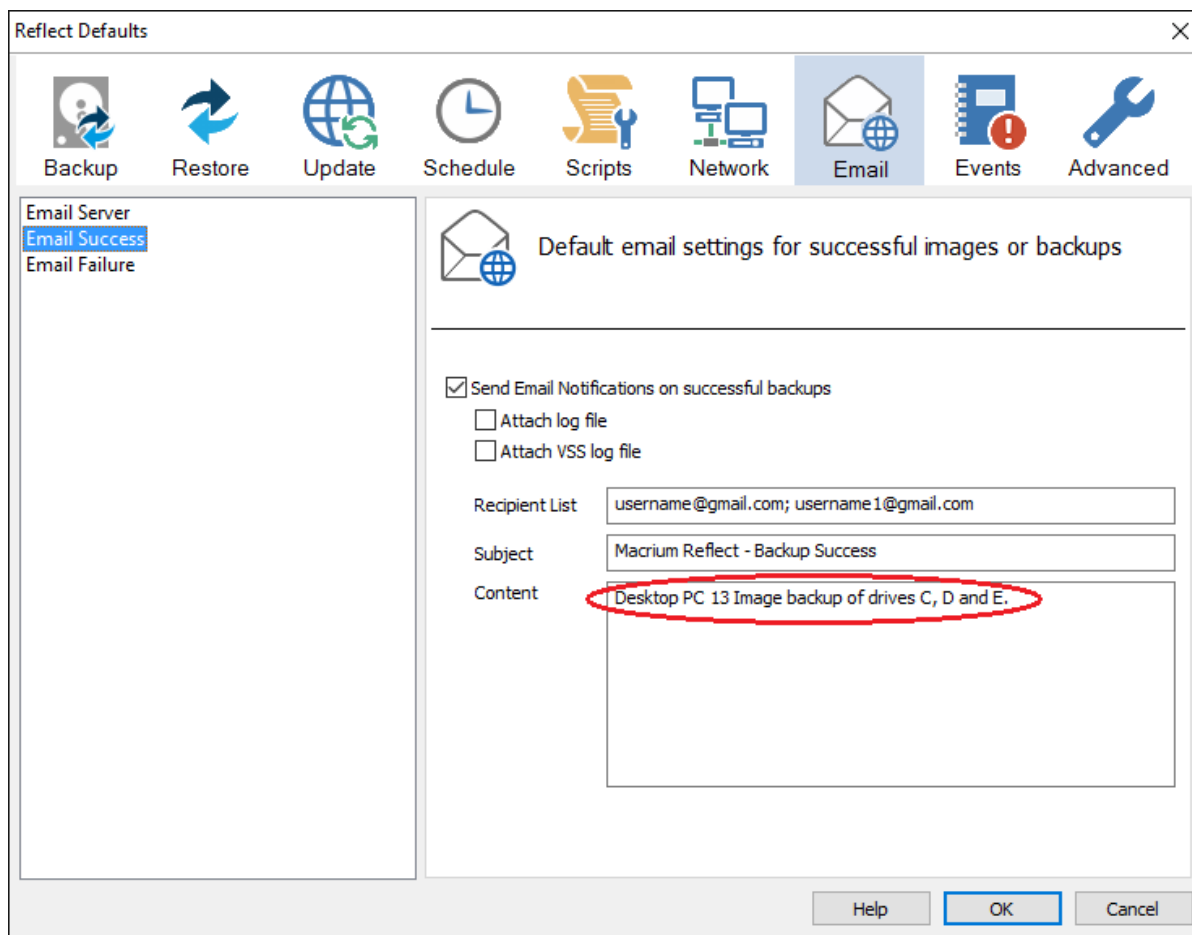
- Click **OK**.

Email failure settings define who receives emails regarding the failure of a backup and what message they receive.

- Select **Email Failure**.

- Select **Send Email Notification on Failed Backups**.

Option	Description
Attach log file	Will attach the log file from the creation of the backup.
Attach VSS log file	Will attach the VSS log with the events made during the backup.

- Enter the email addresses of all recipients in **Recipient List** , separating each email address with a semi-colon.

- Enter the subject for the email in **Subject**.

- Enter a message to be sent regarding the email in **Content**, include the PC that generated the success is identified.
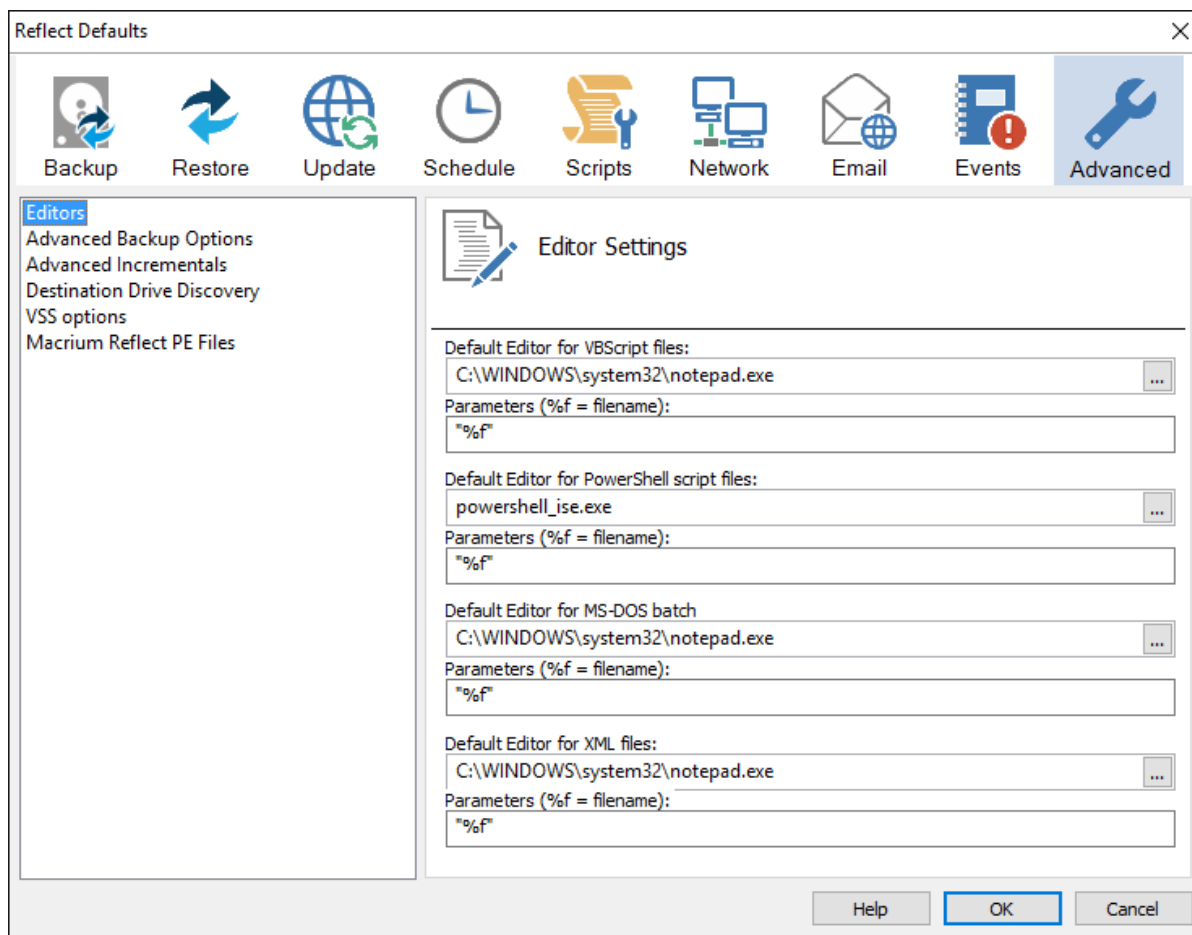
- Click **OK**.

Events

Enable or disable Windows Events generated by Macrium Reflect.
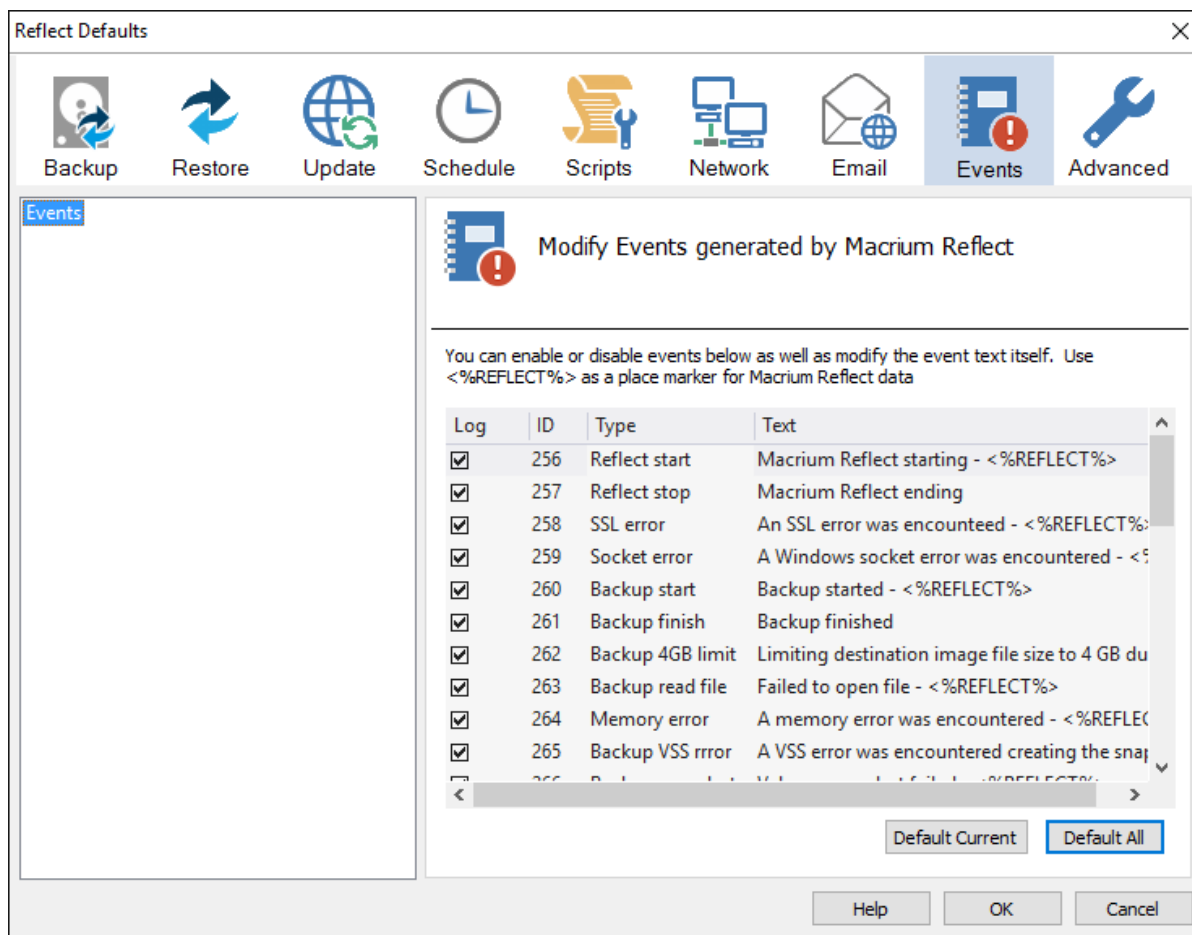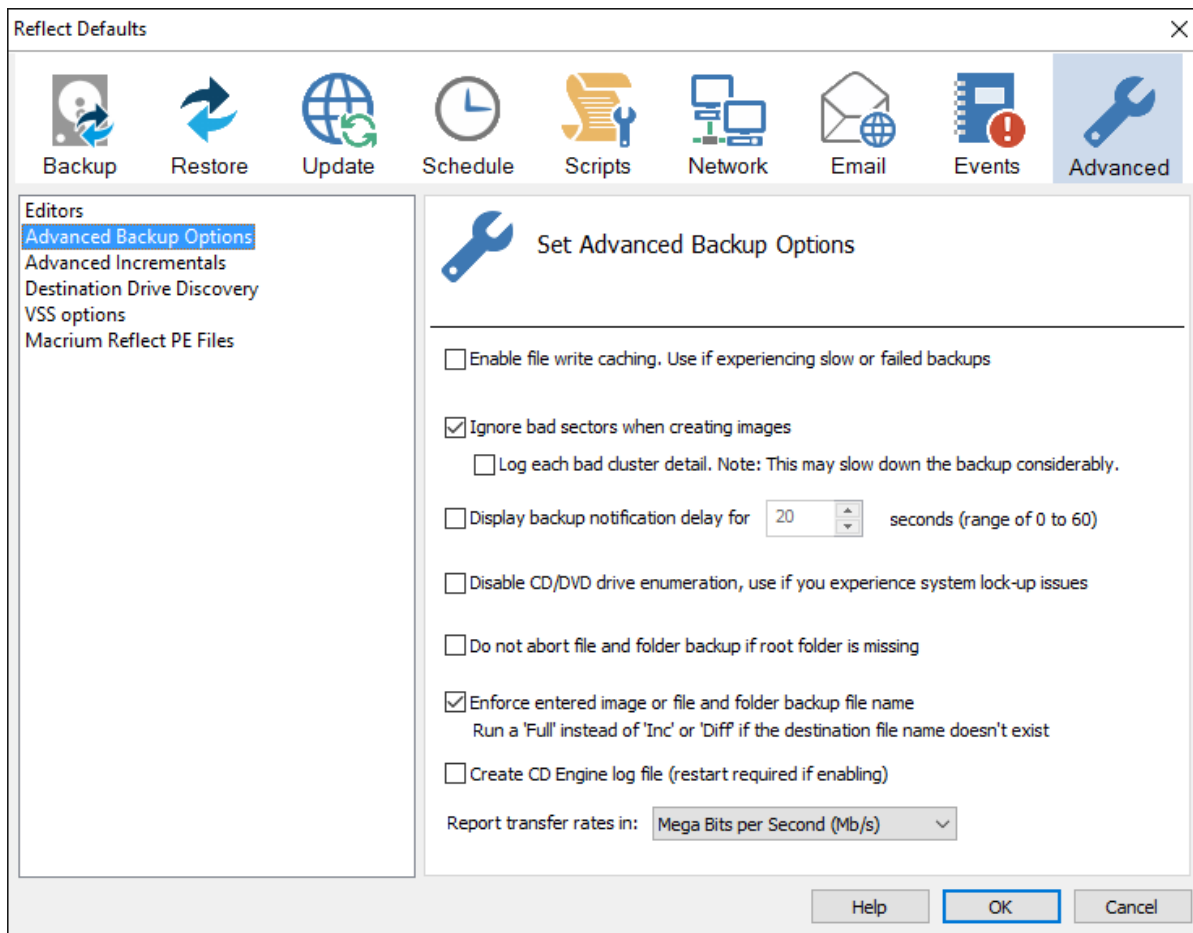
Advanced

Editor settings are used to amend the default editor settings for; VBScript files, Powershell script files, MS-DOS batch and XML.

**Advanced Backup Options** are used to modify your backup options.

Option	Description
Enable file write caching	Is useful if experiencing slow or failed backups.
Ignore bad sectors when creating images.	Damaged disks may still be imaged if this option is selected.
Display backup notification delay for set number of seconds.	Changes the time delay windows is displayed before a scheduled backup starts.
Disable CD/DVD drive enumeration	This option is useful if experiencing system lockup issues.
Do not abort a file and folder backup if a root folder is missing.	Enabling this option will allow your File and Folder backup to continue even if a root folder of the backup is missing.
Enforce an entered image or File and Folder backup file name.	Normally, if you run an Incremental or Differential image of say drive 'C', the target folder is searched for a recent backup set for the same source (Drive 'C'). The backup set is then appended to. The file name you have chosen for the backup will not be used in this matching process. If you enter your own file name and select "Enforce entered image or file and folder backup file name" then only backup sets using the same file name and for the same source will be appended to, if none is found then a new full will be created.
Create a CD Engine log file	Turning this option on allows reflect to create additional logs that can be used for troubleshooting created CD/DVD Rescue Media.

Report transfer rates.	Will allow you to change the way transfer rates are reported in.
	Mega Bits per Second (Mb/s) ⌄ Mega Bits per Second (Mb/s) Mega Bytes per Second (MB/s)

Advanced Incremental options are used for enabling Delta Indexes for Incremental backups
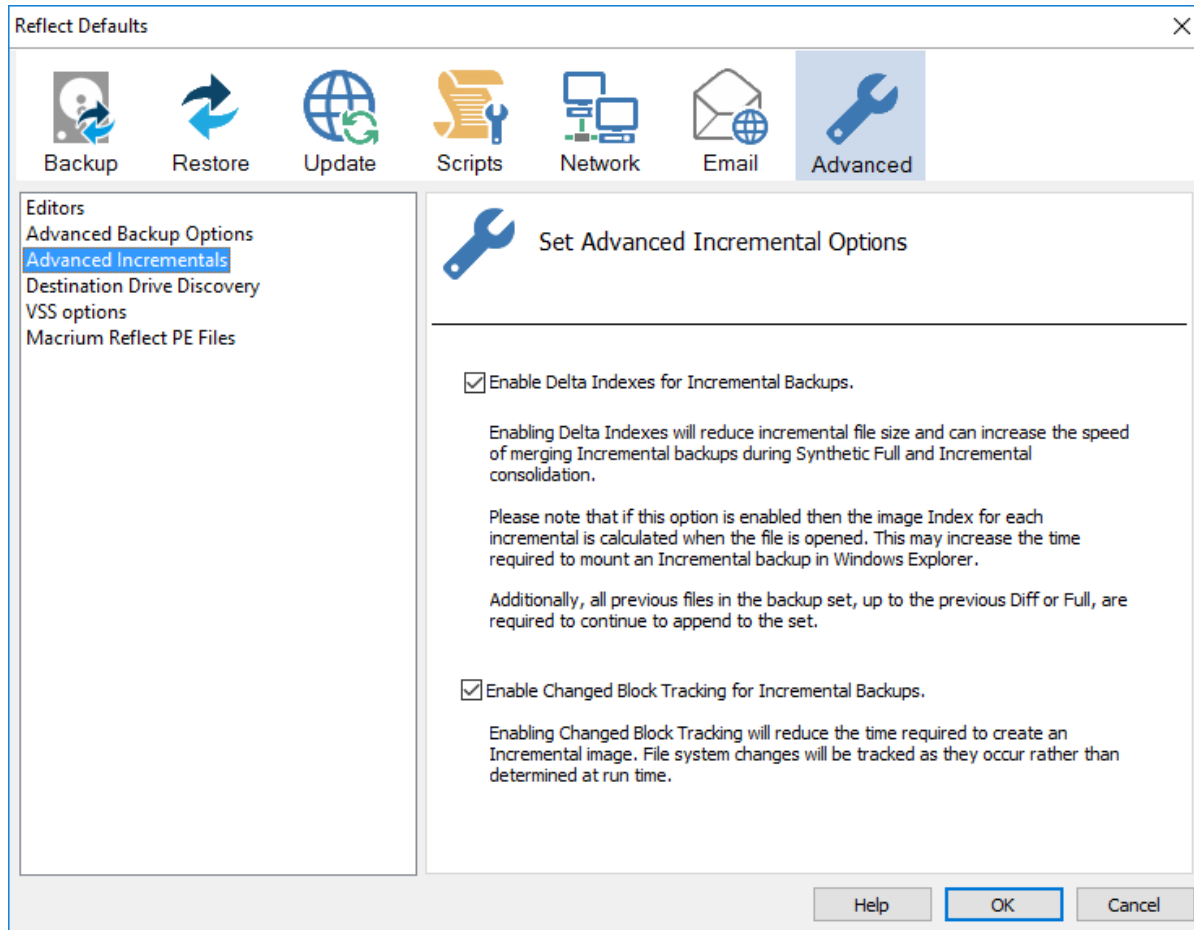

- Enabling **Delta indexes** will reduce incremental file size and can increase the speed of merging Incremental backups during Synthetic Full and Incremental consolidation.
  Additionally, all previous files in the backup set, up to the previous Diff or Full, are required to continue to append to the set.

If this option enabled then the image Index for each incremental is calculated when the file is opened. This may increase the time required to mount an Incremental backup in Windows Explorer.

For information on Changed Block Tracker please see here

Destination Drive Discovery options change the way your destination drive is being discovered if the destination is a local drive.
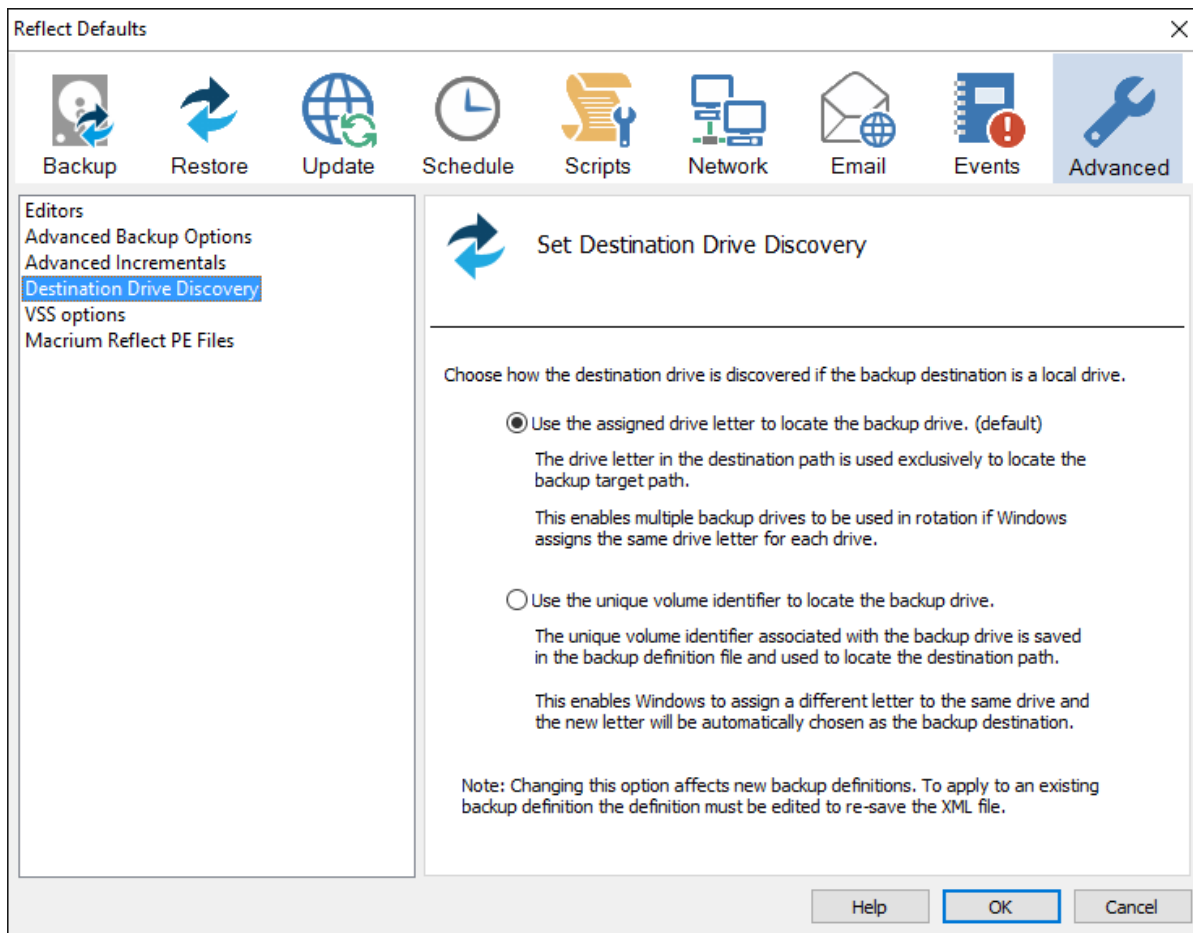
Changing the option affects new backup definitions. To apply to an existing backup definition, the definition XML file needs to be opened and re-saved by clicking 'Finish'.

Option	Description
Use the assigned drive letter to locate the backup drive (default)	The drive letter in the destination path is used exclusively to locate the backup target path. Multiple backup drives can be be used in rotation if Windows assigns the same drive letter for each drive.
Use the unique volume identifier to locate the backup drive	The unique volume identifier is associated with the backup drive is saved in the backup definition file and used to locate the destination path. Windows will assign a different letter to the same dive and the new letter will be automatically chosen as the backup destination.

VSS Options are used to set options for VSS
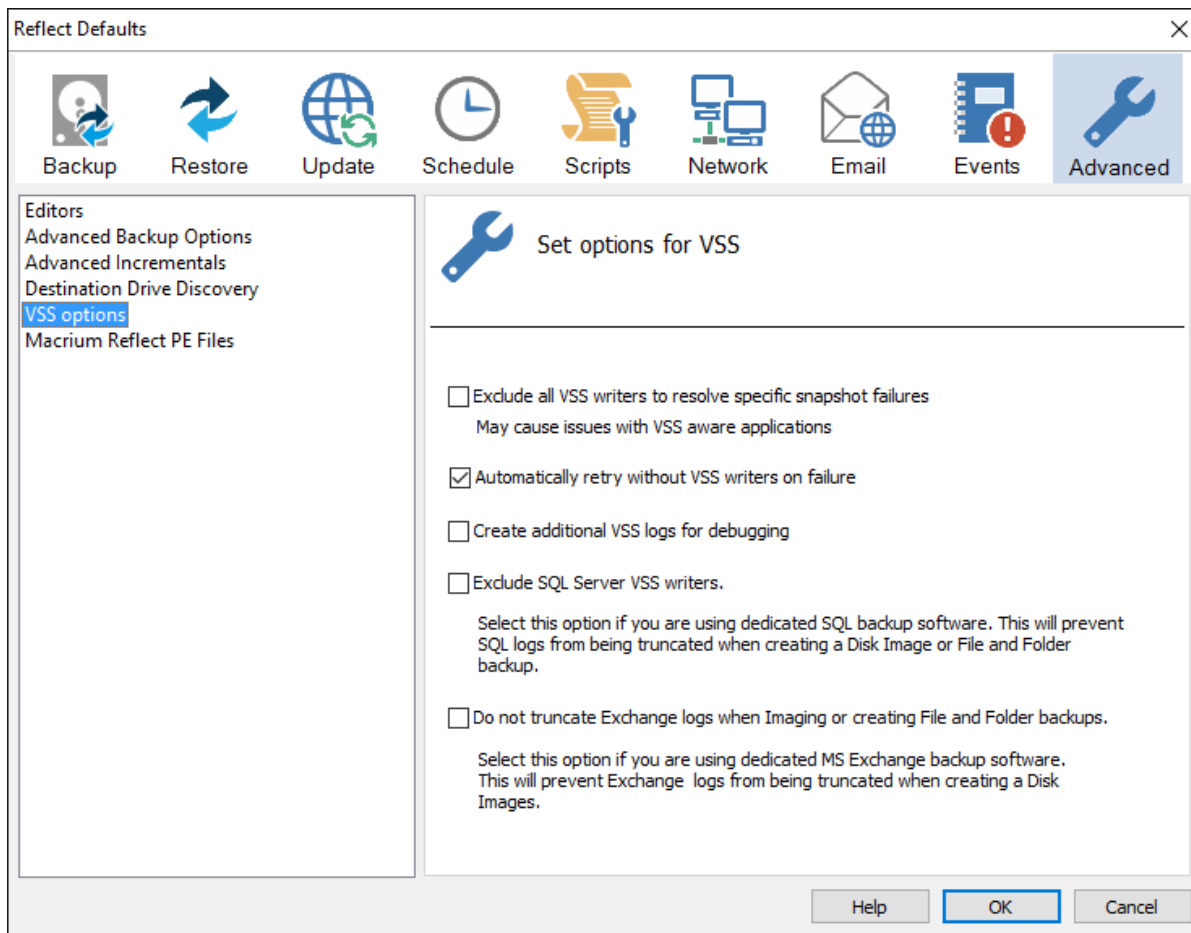
Option	Description
Exclude all VSS writers to resolve specific snapshot failures.	This option will disable VSS writers. Due to Windows system configuration issues, VSS writers are the cause of some backup failures. This may cause issues with other VSS aware applications.
Automatically retry without VSS writers on failure	Reflect will try to create a backup of your system without using VSS; this option is useful when experiencing VSS related errors.
Create additional VSS logs for debugging.	Each backup will create additional VSS logs that can be used in order to troubleshoot VSS related issues when creating backups.
Exclude SQL Server writers.	Select this option if you are using dedicated SQL backup software. This will prevent SQL logs from being truncated when creating a Disk Image or File and folder backup.
Do not truncate Exchange logs when imaging or creating file and folder backups.	Select this option if you are using dedicated MS Exchange backup software. This will prevent Exchange logs from being truncated when creating a Disk Images.

Macrium Reflect PE Files options are used to select the drive where Macrium Reflect PE files will be stored.

Reflect Defaults ✕

| Backup | Restore | Update | Schedule | Scripts | Network | Email | Events | Advanced |

Editors
Advanced Backup Options
Advanced Incrementals
Destination Drive Discovery
VSS options
Macrium Reflect PE Files

Set Options For Macrium Reflect PE Files

Select drive where Macrium Reflect PE files will be stored

Drive	Available Space	File System
☑ C:	9.78 GB	NTFS
☐ D:	22.01 GB	NTFS

Please note. Only valid locations are listed above

| Help | OK | Cancel |

- If your system contains multiple drives you can choose on what Drive Macrium Reflect PE files are saved.