

```
#include <File.au3>
#NoTrayIcon
```

```
#Region.Vars
```

```
; Mail variables
```

```
Global $ScanError
```

```
Global $s_SmtpServer = "smtp.gmail.com"
```

```
Global $s_FromName = "Autolt"
```

```
Global $s_FromAddress = "email@gmail.com"
```

```
Global $s_ToAddress = "email@gmail.com"
```

```
Global $s_Subject = "ScanToCopy Error"
```

```
Global $s_Body = "There was an error using ScanToCopy:" & $ScanError
```

```
Global $s_AttachFiles = ""
```

```
Global $s_CcAddress = ""
```

```
Global $s_BccAddress = ""
```

```
Global $s_Username = "name"
```

```
Global $s_Password = "pwd"
```

```
Global $IPPort = 465
```

```
Global $ssl = 1
```

```
Global $oMyRet[2]
```

```
Global $oMyError = ObjEvent("Autolt.Error", "MyErrFunc")
```

```
Global $ScanError
```

```
; File/folder/watch variables
```

```
Global $szDriveFile, $szDirFile, $szFNameFile, $szExtFile
```

```
Global $FolderName = "Test"
```

```
Global $SourcePath = @HomeDrive & "\" & $FolderName & "\"
```

```
Global $WatchPath = $SourcePath & "*.*"
```

```
Global $TargetPath = @HomeDrive & "\Processed\"
```

```
Global $WatchTime = 20 * 1000 ;Check every 20 seconds
```

```
Global $TimeStamp = TimerInit()
```

```
#EndRegion
```

```
#Region.Watch
```

```
While 1
```

```
  If TimerDiff($TimeStamp) >= $WatchTime Then
```

```
    $FindFile = FileFindFirstFile($WatchPath)
```

```
    If $FindFile <> -1 Then
```

```
      While 1
```

```
        $FoundFile = FileFindNextFile($FindFile)
```

```
        If @error Then ExitLoop
```

```
        $MovedFile = $TargetPath & @MON & "-" & @MDAY & "-" & @YEAR & "-" & @HOUR & "-" & @MIN & "-" & @SEC
```

```
        $MoveIt = FileMove($SourcePath & $FoundFile, $MovedFile, 9)
```

```
        If $MoveIt <> 1 Then
```

```
          $ScanError = $FoundFile & " was not moved."
```

```
          $rc = _INetSmtplibMailCom($s_SmtpServer, $s_FromName, $s_FromAddress, $s_ToAddress, $s_Subject, $s_Body, $s_AttachFiles, $s_CcAddress, $s_BccAddress, $s_Username, $s_Password, $IPPort, $ssl)
```

```
          If @error Then
```

```
            $errorLog = FileWrite($TargetPath & "error.log", "Error:" & @error & " Rc:" & $rc)
```

```
            If $errorLog <> 1 Then
```

```
              MsgBox(0, "Error sending message", "Error code:" & @error & " Rc:" & $rc, 30)
```

```
            EndIf
```

```
          EndIf
```

```
        ExitLoop
```

```
      EndIf
```

```
      Local $PathSplit = _PathSplit($MovedFile, $szDriveFile, $szDirFile, $szFNameFile, $szExtFile)
```

```
      $PathExt = $PathSplit[4]
```

```
      If $PathExt = ".pdf"
```

```
        $PrintIt = _FilePrint($MovedFile)
```

```
        If $PrintIt <> 1 Then
```

```

$ScanError = $FoundFile & " was not printed."
$src = _INetSmtplibMailCom($s_SmtpServer, $s_FromName, $s_FromAddress, $s_ToAddress, $s_Subject)
If @error Then
    $ErrorLog = FileWrite($TargetPath & "error.log", "Error:" & @error & " Rc:" & $rc)
    If $ErrorLog <> 1 Then
        MsgBox(0, "Error sending message", "Error code:" & @error & " Rc:" & $rc, 30)
    EndIf
EndIf
EndIf
EndIf
WEnd
EndIf
$TimeStamp = TimerInit()
EndIf
Sleep (20)
WEnd
#EndRegion

#Region Mail
Func _INetSmtplibMailCom($s_SmtpServer, $s_FromName, $s_FromAddress, $s_ToAddress, $s_Subject = "", $s_Body)
    $objEmail = ObjCreate("CDO.Message")
    $objEmail.From = "" & $s_FromName & "" <' & $s_FromAddress & '>'
    $objEmail.To = $s_ToAddress
    Local $i_Error = 0
    Local $i_Error_description = ""
    If $s_CcAddress <> "" Then $objEmail.Cc = $s_CcAddress
    If $s_BccAddress <> "" Then $objEmail.Bcc = $s_BccAddress
    $objEmail.Subject = $s_Subject
    If StringInStr($s_Body, "<") and StringInStr($s_Body, ">") Then
        $objEmail.HTMLBody = $s_Body
    Else
        $objEmail.Textbody = $s_Body & @CRLF
    EndIf
    If $s_AttachFiles <> "" Then
        Local $S_Files2Attach = StringSplit($s_AttachFiles, ";")
        For $x = 1 To $S_Files2Attach[0]
            $S_Files2Attach[$x] = _PathFull($S_Files2Attach[$x])
            If FileExists($S_Files2Attach[$x]) Then
                $objEmail.AddAttachment($S_Files2Attach[$x])
            Else
                $i_Error_description = $i_Error_description & @If & 'File not found to attach: ' & $S_Files2Attach[$x]
                SetError(1)
                return 0
            EndIf
        Next
    EndIf
    $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/sendusing") = 2
    $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/smtpserver") = $s_SmtpServer
    $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/smtpserverport") = $s_SmtpServerPort
    ;Authenticated SMTP
    If $s_Username <> "" Then
        $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/smtpauthenticate") = 1
        $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/sendusername") = $s_Username
        $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/sendpassword") = $s_Password
    EndIf
    If $s_Ssl Then
        $objEmail.Configuration.Fields.Item ("http://schemas.microsoft.com/cdo/configuration/smtpusessl") = True
    EndIf
EndFunc

```

```

;Update settings
  $ObjEmail.Configuration.Fields.Update
; Sent the Message
  $ObjEmail.Send
  if @error then
    SetError(2)
    return $oMyRet[1]
  EndIf
EndFunc ;==>_INetSmtMailCom
;
;
; Com Error Handler
Func MyErrFunc()
  $HexNumber = Hex($oMyError.number, 8)
  $oMyRet[0] = $HexNumber
  $oMyRet[1] = StringStripWS($oMyError.description,3)
  ConsoleWrite("### COM Error ! Number: " & $HexNumber & " ScriptLine: " & $oMyError.scriptline & " De
  SetError(1); something to check for when this function returns
  Return
EndFunc ;==>MyErrFunc
#EndRegion

```