

## Application Note

# Calling WaveMail from an application or a batch-file to send a file as attachment

## CONTENTS

<b>1. Overview .....</b>	<b>1</b>
<b>2. Calling WaveMail.....</b>	<b>2</b>
2.1. Parameters.....	2
2.1.1. Necessary parameters.....	2
2.1.2. Optional parameters .....	2
2.2. Examples.....	3
2.2.1. Example 1.....	3
2.2.2. Example 2.....	3
2.3. Remarks .....	3
2.4. Example of a call implemented in Delphi .....	4

## 1. Overview

---

Starting with WaveMail version 2.01 it is possible to call WaveMail from an application or from a batch-file in order to send a file as an attachment. The filename is passed as a command line parameter when calling WaveMail.

This feature helps to automate the process of sending an attachment. Sending an attachment is normally done by manually creating a new message and selecting the file to attach in the 'Add Attachment' dialog. This is quite easy, however if this procedure has to be executed very often, it might be easier to automate this process.

Application examples:

- An unmanned remote site is equipped with some measurement equipment. This equipment is PC-based. An application on the PC creates a daily data file which has to be transferred to the headquarter (without manual intervention on the remote site). The system can be configured such that WaveMail will be called everytime a data file is complete.
- A PC-based encryption software creates cyphertext output which has to be sent over HF. Instead of manually creating a new WaveMail message and selecting the cyphertext in the 'Add Attachment' dialog, the encryption software calls WaveMail with the filename of the cyphertext as parameter.

## 2. Calling WaveMail

---

### 2.1. Parameters

Lets start with an example:

```
c:\programs\wavemail\program\wavemail.exe sendfile=c:\temp\anydatafile.dat
```

This is a call with just one parameter `sendfile=c:\temp\anydatafile.dat` . The `sendfile` parameter is always needed. Other parameters are optional if an operator will manually complete the message with the destination address and subject. If the message with the attached file should be sent without any user intervention, then additional parameters are necessary (see below).

#### 2.1.1. Necessary parameters

##### **sendfile=**

Use this parameter to define the file to be attached. Enter the full path.

Example: `sendfile=c:\temp\anydatafile.dat`

#### 2.1.2. Optional parameters

##### **to=**

This parameter defines the recipient(s). Recipients must be seperated by semicolons ';'.

Example: `to=user5.stationxy;first.last@company.com`

##### **cc=**

This parameter defines the CC recipient(s). Recipients must be seperated by semicolons ';'.

Example: `cc=user99.stationxy`

##### **subj=**

This parameter defines the subject line of the mail. The subject text must be enclosed in double quotes .

Example: `subj="This is a test"`

##### **user=**

This parameter defines the username for automatic login at WaveMail. If local users are defined in WaveMail, WaveMail will only create the mail with the attachment after a user is logged in. You can perform the login procedure manually (normal login) or you can pass the username and password as parameters when calling WaveMail in order to automate the login process. (Note: the user will stay logged in after the message has been created).

Example: `user=user1`

##### **pass=**

See parameter `user=` .

Example: `pass=mypassword`

##### **noprompt**

After WaveMail has created the message with the attached file, the user has to activate the 'Send' button in the 'New Message' window in order to send the message. If the keyword `noprompt` is found in the command line, WaveMail will immediately put the message into the send-queue without any user intervention. However this can only work, if at least one recipient is defined with the parameter `to=` , because otherwise WaveMail does not know where to send the mail.

## 2.2. Examples

### 2.2.1. Example 1

Call:

```
c:\programs\wavemail\program\wavemail.exe sendfile=c:\temp\anydatafile.dat
```

Action:

- If WaveMail is not yet running when the call is made:  
WaveMail is started. If no local users are defined in WaveMail, then WaveMail will start with the Postmaster automatically logged in. In this case the 'New Message' window will open with the desired file in the attachment list and the user can enter the recipient(s), subject and text body. Then the user can send the message.  
If local users are defined in WaveMail, then WaveMail will start and the 'Login' dialog will be displayed. Once a user logs in, the 'New Message' window will appear and the user can enter the recipient(s), subject and text body. Then the user can send the message.
- If WaveMail is already running and a user is logged in (if no local users are defined, then the Postmaster is automatically logged in):  
The 'New Message' window will open with the desired file in the attachment list and the user can enter the recipient(s), subject and text body. Then the user can send the message.

### 2.2.2. Example 2

Call:

```
c:\programs\wavemail\program\wavemail.exe sendfile=c:\temp\anydatafile.dat to=stationxy  
subj="measurement data 25.05.2005" user=user1 pass=mypassword noprompt
```

*Note: the complete call is on one line, here the command is only split due to the size limit.*

Action:

- WaveMail is started (if not already running), the login procedure for user1 will be executed (if not yet logged in) and a new message with recipient `stationxy` is created. The file is attached to the message, the subject `measurement data 25.05.2005` is added and the message is sent (put into the send queue).  
(If a different user is currently logged in, then the request will only be handled after that user logs out.)

## 2.3. Remarks

- If a call to WaveMail is made, but no user is logged in and no `user=` and `pass=` parameters are passed to WaveMail, then the request will stay in an internal queue until a user logs in. If WaveMail is terminated before a user logs in, then the request to create a message with the desired attachment will be lost.
- WaveMail handles the request one after the other. Therefore if a call to WaveMail is made without `user=` and `pass=` parameters (and no user is logged in on WaveMail), then any following call (even with correct `user=` and `pass=` parameters) will not be processed until a user logs in and completes the message generated by the first call.
- If several calls to WaveMail are made without `user=` and `pass=` parameters (and no user is logged in on WaveMail), then all requests will be handled one after the other once a user logs in.
- If the 'New Message' window is already open when a call to WaveMail is made, then the 'New Message' window is brought to the top (if not already on top) and the request is only processed after the current 'New Message' is sent or cancelled.
- If the `user=` parameter is passed to WaveMail then the request will only be handled once the specified user is logged in. If a different user is currently logged in, then the request is only handled after the current user logs out and the desired user logs in (either automatically if both `user=` and `pass=` parameters are passed in the call or manually if the `pass=` parameter is missing or not correct).

## 2.4. Example of a call implemented in Delphi

```
procedure SendMail;
var StartupInfo: TStartupInfo;
    ProcessInfo: TProcessInformation;
    Res:dword;
    em:shortstring;
begin
  { Prepare StartupInfo }
  FillChar(StartupInfo,SizeOf(StartupInfo),0);
  with StartupInfo do
  begin
    cb:= SizeOf(StartupInfo);
    dwFlags:= Startf_UseShowWindow;
    wShowWindow:= SW_NORMAL;
  end;
  { Call WaveMail }
  if not CreateProcess(nil,'C:\programs\wavemail\program\wavemail.exe sendfile=c:\temp\file1.txt',
    nil,nil,False,Normal_Priority_Class,nil,nil,StartupInfo,ProcessInfo)
  then
  begin
    { Error handling }
    Res:=GetLastError;
    fillchar(em,sizeof(em),0);
    FormatMessage(FORMAT_MESSAGE_FROM_SYSTEM,
      nil,Res,0,@em[1],255,nil);
    showmessage(PChar(@em[1]));
  end;
end;
```