

Join SMS FoxBox with POSTFIX Mail Transfer Agent

As we discussed in the article [How to join SMS FoxBox to your server environment](#) is really simple integrate SMS FoxBox with any kind of Server.

In this article I will explain how to extend the capabilities of a Mail Server.

With the software available below you should be able of sending SMS messages to any mobile phone simply writing an e-mail addressed to PHONE_NUMBER@YOUR_DOMAIN.XXX with the text of the e-mail as body of the message.

Below is listed all what you need to enable this functionality:

- A server running Postfix as MTA
- A SMS FoxBox
- 2 Shell scripts
- Some simple configuration issues on the mail server

How the script works, postfix content filtering

Postfix is a MTA, it is widely used and it is really simple to configure.

It is included for example in Fedora[®], Mac OS X[®] and many others Linux and BSD distributions.

The feature that we are going to implement, is based on Postfix After-Queue Content Filter.

The main difference with the other tutorials on Postfix website is that we are going to use the "filter" as a "wrapper" for SMS FoxBox.

To perform this action we need a script that each time we transfer a message it verify if we are going to send an SMS, and in this case it will launch the URL of SMS FoxBox, with all parameters.

The script is reported below:

```
#!/bin/sh
#
### Script: SMS FoxBox Wrapper for postfix
#
# Author: Davide Cantaluppi
# Platform: Not platform dependent
#
# Purpose: This script can be applied as filter to postfix http://www.postfix.net/FILTER\_README.html
# and using a regexp if the Recipient is a number of 10 or more digits it try to divert the
# message content as SMS using SMS FoxBox as Gateway
#
# DEPENDENCIES: The script requires
# - LYNX
# - FORMAL command that is part of PROCMAIL
# - URLENCODE.sh shell script by Heiner Steven (heiner.steven@odn.de)
#
### CONFIGURATION
#
#Configure these PATH to match your system structure
#
#Localize these. The -G option does nothing before Postfix 2.3.
#specify a path suitable for SMS operations
INSPECT_DIR=/var/spool/sms
#PATH to SENDMAIL command
SENDMAIL="/usr/sbin/sendmail -G -i" # NEVER NEVER NEVER use "-t" here.
#PATH to LYNX command
LYNX="/Applications/lynx.command"
#PATH to formail command
FORM="/usr/bin/formail"
#PATH TO URLENCODE SCRIPT
URLENCODE=/urlencode.sh
# E-Mail address enabled to divert e-mail to SMS only this address is enabled to send SMS
SMSUSR="kanta@kdev.it"
FILEL='/var/log/rcpt.log'
FILEM='/var/log/msgl.log'
#
#
#
###BEGIN OF PROCEDURE
```

```

#
FROM="$1"
RCPT="$@"
# Exit codes from
EX_TEMPFAIL=75
EX_UNAVAILABLE=69
# Clean up when done or when aborting.
trap "rm -f in.$$" 0 1 2 3 15
# Start processing.
cd $INSPECT_DIR || {
echo $INSPECT_DIR does not exist; exit $EX_TEMPFAIL; }
cat >in.$$ || {
echo Cannot save mail to file; exit $EX_TEMPFAIL; }
# PROCEDURE TO LAUNCH THE SMS SENDING
if [ "$FROM" = "$SMSUSR" ]; then
ext_rcpt=`echo $RCPT > $FILEL`
# EXTRACT THE PHONE NUMBER
PHONE=`awk -F @ '{print $1}' $FILEL`
ISPH=`echo $PHONE | grep '[0-9]{10}`
if [ -z "$ISPH" ]; then
$SENDMAIL "$@" $FILEM`
TXTF=`$URLENCODE $FILEM`
#PREPARE THE URL FOR SMS SENDING
LINKG='http://192.168.7.21/source/send_sms.php?username=Admin&pwd=password&from='$FROM'
&nphone='$PHONE'&testo='$TXTF
#Debugging info
#echo $LINKG
#
#SEND THE MESSAGE AS SMS
CMDS=`$LYNX -dump $LINKG`
echo Message sented as SMS; exit 0;
fi
$SENDMAIL "$@"

```

As you can see the script requires some external software that normally is already installed on the system:

- LYNX The command line browser
- FORMAIL command that is part of PROCMail
- URLENCODE.sh shell script by Heiner Steven (heiner.steven@odn.de) to perform TEXT to URL encoding transforms included in the source Archive

If your system meet the pre-requirements you can install the script simply following these steps:

- Download the archive containing filter.sh and urlencode.sh and put these 2 scripts in /usr/sbin.
- Give execution privileges to the script with chmod 0333 script_name.sh for both scripts.
- Change paths in configuration section of filter.sh to your location on the server, or create the folder needed: -
INSPECT_DIR=/var/spool/sms create it with (*) mkdir -p /var/spool/sms
- SENDMAIL="/usr/sbin/sendmail -G -i" Path to the sendmail command
- LYNX="/usr/local/bin/lynx" Path to lynx command
- FORM="/usr/bin/formail" Path to formail command
- URLENCODE=/usr/sbin/urlencode.sh Position of urlencode.sh script
- SMSUSR="kanta@kdev.it" From e-mail enabled to E-mail to SMS divert
- FILEL='/var/log/rcpt.log' Log file create it with touch /var/log/rcpt.log
- FILEM='/var/log/msgl.log' Log file create it with touch /var/log/msgl.log

Then you can proceed as described in the Postfix After-Queue Content Filter how-to:

- Create a dedicated local user account called "filter". This user handles all potentially dangerous mail content - that is why it should be a separate account.
 - Do not use "nobody", and most certainly do not use "root" or "postfix".
 - Create a directory /var/spool/sms that is accessible only to the "filter" user. This is where the content filtering script is supposed to store its temporary files. (*) mkdir -p /var/spool/sms
 - Configure Postfix to deliver mail to the content filter with the pipe(8) delivery agent. /etc/postfix/master.cf:
- ```

=====

```

```
service type private unpriv chroot wakeup maxproc command
(yes) (yes) (yes) (never) (100)
=====
filter unix - n n - 1 pipe
flags=Rq user=filter argv=/usr/sbin/filter.sh -f ${sender} -- ${recipient}
```

To turn on content filtering for mail arriving via SMTP only, append "-o content\_filter=filter:" to the master.cf entry that defines the Postfix SMTP server: /etc/postfix/master.cf:

```
=====
service type private unpriv chroot wakeup maxproc command
(yes) (yes) (yes) (never) (100)
=====
smtp inet ...other stuff here, do not change... smtpd
-o content_filter=filter:
```

- The "content\_filter" line causes Postfix to add one content filter request record to each incoming mail message, with content "filter:dummy". This record overrides the normal mail routing and causes mail to be given to the content filter instead.
- The content\_filter configuration parameter accepts the same syntax as the right-hand side in a Postfix transport table.
- Execute "postfix reload" to complete the change.