



FUNAMBOL mobile open source

Funambol Email Connector & InboxListener Quick Start Guide

Table of Contents

1. Introduction.....	3
1.1. Release Notes.....	3
1.2. Known Issues.....	3
1.3. Related Documents.....	4
2. Funambol's Email Synchronization Environment.....	5
2.1. Environment Description.....	5
2.2. Installation Steps.....	6
3. Funambol's Email Synchronization Installation Procedure.....	8
3.1. Installing Funambol Email Connector.....	8
3.2. Installing Funambol InboxListener.....	9
3.3. Configuring Email Connector.....	10
3.3.1. Handling Mail Server.....	11
3.3.2. Handling Account.....	12
3.3.3. Configuring the Email Connector using the Command Line Tool.....	15
3.4. Configuring Email SyncSource.....	16
3.5. Officer Configuration.....	17
3.6. Encryption.....	18
3.7. Enabling Log.....	19
3.7.1. Enabling log in the Email Connector.....	19
3.7.2. Enabling log in the InboxListener.....	20
4. Appendix A - Notification System between Funambol DS-Server and Mobile Device.....	21
4.1. Mobile Device Settings.....	22
4.2. Admin Tool Settings.....	23
5. Appendix B – Mail Server Configuration.....	26
5.1. Microsoft Exchange 2000/2003.....	26
5.2. Domino 6.5/7.0.....	26
5.3. GMail.....	27
5.4. Yahoo.....	28
5.5. Courier (Horde Web Access).....	28
5.6. CommuniGate Server 5.x.....	30

1. Introduction

The purpose of this document is to describe how to manage and administer the Funambol Email Connector and the Inbox-Listener component. The admin user will use the FunambolAdmin console and a command line tool in order to configure the entire Funambol's email synchronization environment.

This document is intended to be read by the administrative users.

1.1. Release Notes

The current version of the email connector is compliant with the following clients:

- funambol windowmobile/pocketpc plugin
- funambol smartphone plugin
- funambol blackberry plugin
- funambol j2me email client for mobile phone
- Nokia E60 (prototype)

The current version was tested with:

- Ms Exchange 2000/2003 Server
- Lotus Domino 7.0
- Zimbra Collaboration Suite
- CommuniGate Server 5.x
- Some unix/linux based Mail Server (Cyrus, Courier, ...)
- Some MS-Windows based Mail Server (Winmail, Mercury, ...)
- Some public Email Service Provider (Gmail, Yahoo, AOL, ...)

1.2. Known Issues

This is a list of some of known issues about the Funambol Email Connector:

- “forwarded” flag for the email in the inbox folder is not supported yet
- in the CommuniGate Server we have to specify as inbox folder name 'INBOX'. At the moment to do this operation the admin must insert the value in the directly in the DB schema (see appendix)
- the jpg attachment send by Microsoft Outlook 2002 client (MS Outlook XP) is not correctly downloaded and handling by the Email Connector
- some email with the body in html format has the content too complex for the HtmlParser in the email connector. We should keep simple the parser in order to improve the performance, but in this case some html tag can be present in the email on the device.
- ...

1.3. Related Documents

The following documents are related to this design document:

- [1] DS Modules Development Tutorial
- [2] DS Server Administration Guide

2. Funambol's Email Synchronization Environment

2.1. Environment Description

Before starting with the installation procedure description we have to provide a quick explanation about the Funambol's email synchronization environment.

To implement the push email functionality and improve the performance, we start with the *Inbox-Listener Tool*. We configure this component to provide the following functionality:

- polling a Mail Server for new mail for a specific email account; when new mail has been received for that account the tool sends a notification to the DS server (inbox-listener feature)
- storing some basic email information (i.e. Message-ID, Date, ...) from the emails in the inbox folder into a local caching system in order to improve the synchronization performance. The user can define how many email the system should store in the caching system. (message processor feature)

To implement the email synchronization feature we have to install in the *Email Connector* in the DS Server. We configure this component to provide the following functionality:

- synchronize the email for a user between a Mail Server and a Mobile Device

The email synchronization process is summarized below Figure 1:

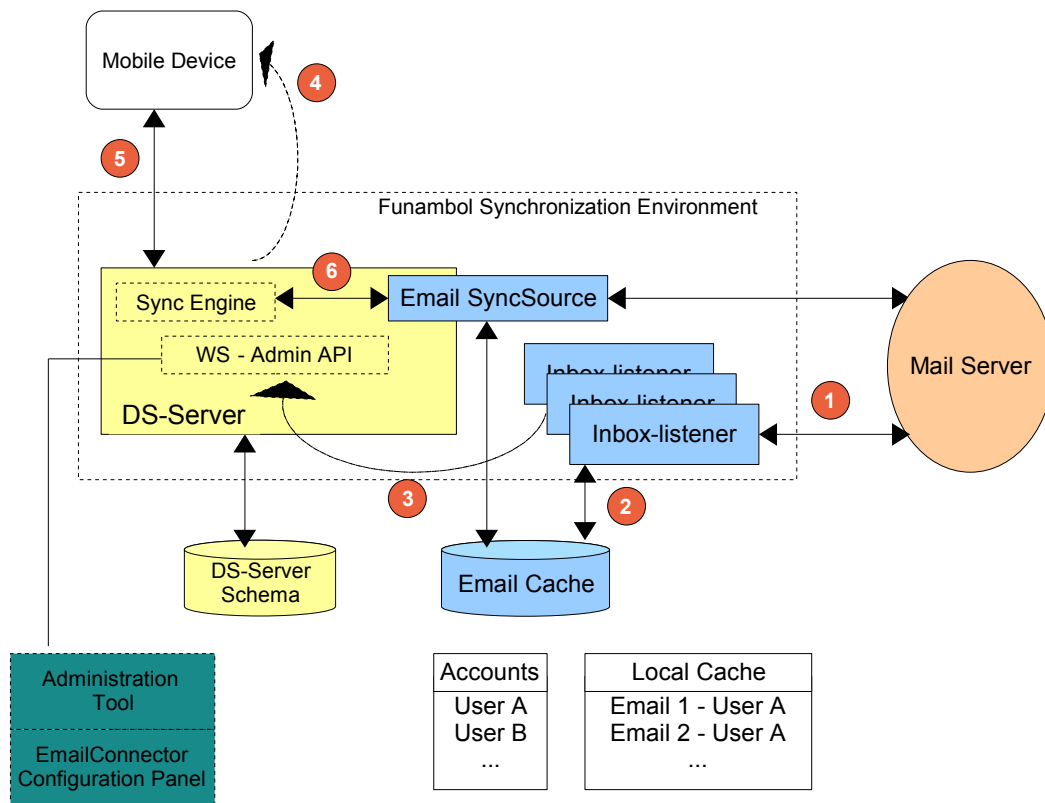


Figure 1: Funambol email synchronization process

1. The *Inbox Listener* checks for updates on a user inbox (it reads the users to sync from "Accounts");
2. the *Inbox Listener* updates the Email cache (It gets 'N' emails from the Server Inbox Folder and stores some basic info in the "Local Cache". This behaviour allows the Email Connector to improve the performance during the sync process); if there is a new email, it sends a notification to the DS Server;
3. the *Inbox Listener* tells the DS Server to notify the user that there is a new email;
4. the DS Server sends a notification message to the user's device
5. the *Mobile Device* starts a new sync session. Note that the *Mobile Device* can start sync session in two ways: 1.driven by a DS server Notification. The DS Server sends a notification if a new email is in the inbox folder. 2. driven by the user. The user starts the sync session pressing the 'sync button' on the syncML plug-in
6. the DS Server's Synchronization engine calls the *Email SyncSource* in order to synchronize the user's inbox; note that the Email SyncSource accesses both the Email Cache first, and than the Mail server only if really needed.

2.2. Installation Steps

In this paragraph we give a short list of the main steps in order to install the Funambol Email Connector & Inbox Listener:

- Check if there is an available DB connection (postgres or hypersonic)
- Install the Funambol Email Connector
 - the installation automatically creates the Connector DB schema.

- Install the Inbox-Listener Tool
 - configure the InboxListener.xml and shell script
- Run the DS Server
- Run the Admin Tool
- Configure the Email Connector Properties
 - MODE 1 : set the Accounts in the Email Connector Panel in the Admin Tool (in order to get the IMAP Folder Information the administrator can use an option of the Inbox-Listener *Command Line Tool*)
 - MODE 2 : set the Accounts using the Inbox-Listener *Command Line Tool*
- Check the Email SyncSources
 - create and set the parameters in the Email SyncSource Panel in the Admin Tool
- [optional] Configure the Officer in the Server Settings Panel in the Admin Tool
- [optional] Set the Encryption in the Email SyncSource Panel in the Admin Tool
- [optional] Set the log Level
- set the “push environment” information in the Device Settings Panel (see ch. 4)
- Run the *Inbox-Listener Tool*
 - the Message-Processor loads the inbox folder emails in the cache table (just basic email info; NOT the entire email)
 - the Inbox-Listener starts polling the inbox folder.
- Get the Mobile Device and run a sync session from the Mobile Device or wait for a notification from the DS-Server

3. Funambol's Email Synchronization Installation Procedure

3.1. Installing Funambol Email Connector

The Funambol Email Connector is distributed as a standard Funambol module [1]. The distribution contains the following files:

- funambol-email-connector-<major>.<minor>.<buildnumber>.s4j
- the Inbox-Listener Tool
- this guide

To install the module you have to follow this steps:

1) put the s4j file in the directory

`<installation dir>\ds-server\modules`

2) modify “install.properties” file adding “funambol-email-connector-x.x.x” to the modules list:

`modules-to-install=foundation-3.0.0,pdi-1.5,pimweb-1.1,funambol-enterprise-activation-3.x.x,funambol-email-x.x.x`

3) start installation modules command.

For more details about the Funambol module installation see [2].

During the installation the following steps are performed automatically:

1. the database is initialized creating the connector specific tables and registering the connector into the server.
2. the EmailOfficer.xml file is copied in the directory:
`<installation dir>\ds-server\config\funambol\server\security`

Note: before running the Funambol DS-Server you have to put the JDBC in the Application Server library folder; for instance in a Tomcat + Postgres installation you have to copy the postgresql-***.jdbc3.jar library in the “<tomcat_home>\common\lib” folder (just for brand new installation)

Note: the JavaMail library is already included in the distribution so that you don't have copy the mail.jar and the activation.jar in the Application Server library folder.

3.2. Installing Funambol InboxListener

The Funambol InboxListener tool is included in the Funambol Email Connector archive file:

– funambol-email-<major>.<minor>.<buildnumber>.zip

To install the tool you have to follow this steps:

1) unzip the “.zip” and check the directory structure:

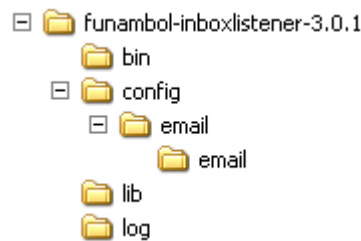


Figure 2: inbox listener home

2) modify the run script: inboxlistener.* ; change JAVA_HOME and JDBC_JAR according to your environment (in the bin directory)

3) modify “email/ema/InboxListener.xml” according with your environment

<i>Property</i>	<i>Description</i>
listnerid	Inbox-Listerner module info; listener id linked to the mailboxes
startupNotification	If true when the module starts, the inbox-listener sends a notification for all the registered user to the sync server
saveSubject	If true the inboxlistener saves the subject in the caching system (for debug purpose)
saveSender	If true the inboxlistener saves the sender in the caching system (for debug purpose)
reloadingTime	Every “x” minutes the Inbox-Listener checks the list of accounts in the DB
keystoreHome	Path where is stored the keystore file
driverClassName	i.e. org.postgresql.Driver
url	i.e. jdbc:postgresql://localhost:5432/funambol_600
username	Database username
password	Database password
maxActive	Max number of connection in the db connection pool

<i>Property</i>	<i>Description</i>
wsEndPoint	i.e. http://localhost:8080/funambol/services/admin
wsUsername	Funambol Administrator user
wsPassword	Funambol Administrator password
wsSyncSource	Name of the Email SyncSource

4) Run the inbox-listener with the following procedure:

- change directory : <ds-server home>/inbox-listener
- type command ./inboxlistener.sh start

Note: there are 4 option in order to handle the inbox-listener tool

- > ./inboxlistener.sh start : start the inboxlistener
- > ./inboxlistener.sh stop: stop the inboxlistener
- > ./inboxlistener.sh start -debug : start the inboxlistener with console debug
- > ./inboxlistener.sh console : start the command line tool. (see 3.3.3 paragraph)

3.3. Configuring Email Connector

In order to configure the Funambol Email Connector the administrator can open the Funambol Administrator Console and browse in the following tree structure (Figure 3)



Figure 3: Email Connector Access Point

When the administrator points on the Email Connector item he can see the following panel (Figure 4).

Public Mail Servers

Server description

Start with ▼

Reset

Search

Description	Outgoing server	port	Incoming server	port

Add

Edit

Delete

Accounts

Username

Start with ▼

Reset

Search

Server description

Start with ▼

Username	Login	Address	Activation	Max email	Delete o...	Descript...	Protocol

Add

Edit

Delete

Figure 4: Email Connector Configuration First Panel

In the first table there is the Public Mail Servers list (i.e. Gmail, Yhadoo, ...)

In the second table there is the Accounts list for the registered users in the Funambol DS-Server.

3.3.1. Handling Mail Server

In order to create a new Mail Server the administrator must press the “ADD” button in the “Mail Server section” (Figure 4).

A new panel is opened (Figure 5). In this panel the administrator can add all the settings about a mail server

Description	<input type="text"/>			
Server product	Other			
Protocol	imap			
Outgoing server	<input type="text"/>	Port	<input type="text" value="25"/>	Auth <input type="checkbox"/>
Incoming server	<input type="text"/>	Port	<input type="text" value="143"/>	
SSL Outgoing	<input type="checkbox"/>			
SSL Incoming	<input type="checkbox"/>			
Key store filename	<input type="text"/>			
Key store passphrase	<input type="text"/>			
Soft-Delete on server	<input type="checkbox"/>			
Inbox folder name	<input checked="" type="checkbox"/>	<input type="text" value="Inbox"/>		
Outbox folder name	<input checked="" type="checkbox"/>	<input type="text" value="Outbox"/>		
Sent folder name	<input type="checkbox"/>	<input type="text" value="Sent"/>		
Draft folder name	<input type="checkbox"/>	<input type="text" value="Drafts"/>		
Trash folder name	<input type="checkbox"/>	<input type="text" value="Trash"/>		
Save		Close		

Figure 5: Add mail server panel

3.3.2. Handling Account

In order to create a new account for a Funambol DS-Server user the administrator must press the "ADD" button in the "users section" (Figure 4).

A new pop-up panel is opened. In this panel the administrator can search the Funambol DS-Server user to configure (Figure 6).

Search Users

Username: Start with Search

First Name: Start with Reset

Last Name: Start with

E-mail: Start with

Username	First Name	Last Name	E-mail
----------	------------	-----------	--------

Figure 6: Funambol DS-Server User Selection

When the Administrator press search the table will show all the ds-server users.
When a user is selected the “account settings panel” (Figure 7) is opened

Login:

Password:

Confirm password:

E-mail address:

Listener Id:

Active: ☒

Push: ☒

Refresh time (min):

Max email number:

Max IMAP emails:

Mail server: custom

Description:

Server product: Other

Protocol: imap

Outgoing server: Port: Auth: ☐

Incoming server: Port:

SSL Outgoing: ☐

SSL Incoming: ☐

Key store filename:

Key store passphrase:

Soft-Delete on server: ☐

Inbox folder name: ☒

Outbox folder name: ☒

Sent folder name: ☐

Draft folder name: ☐

Trash folder name: ☐

Save Close

Figure 7: Account Settings Panel

In this panel the administrator can set the parameters like in the table below:

left form

<i>Property</i>	<i>Description</i>
username	Mail server login
Password / confirm password	Mail server password
Email address	User email address
listenerID	Listener ID (see inbox-listener documentation)
active	If the user is active
push	If the push environment is active for the specific user
Refresh time	Every "x" minutes the Inbox-Listener tool refreshes the Caching system
Max Inbox Email	Max number of emails in the caching system
Max IMAP Email	Max number of emails that will be synchronized in the Drafts, Sent, Trash folder

right form

<i>Property</i>	<i>Description</i>
Mail Server	The administrator can choose either a public Mail Server or the <custom> option. If the administrator chooses the <custom> option he must insert all the mail server information
Description	A brief description of the mail server
Server Product	Define the mail server product: i.e. Exchange, Courier, Other
Protocol	Used protocol in the sync sessions
Out Server / Out Port / Auth.	All the information about the outgoing server.
In Server / In Port	All the information about the incoming server.
SSL outgoing	SSL Activation for the outgoing server
SSL incoming	SSL Activation for the incoming server
Keystore path	Path of the java keystore
Keystore password	Password of the java keystore
Soft-Delete on Server	Enable the soft delete on the Funambol DS-Server
*** Folder	Activation and information about the path folder (the complete name of the folder)

Folder configuration

Note: the Funambol plug-in and SyncML email client can synchronize the following folders:

- inbox
- outbox

So in the Account panel the admin must check just this two folder.

In the pop-based account this is already done and the folder configuration is disable.

In the imap-based account the folder configuration is enable but the admin user must set just the inbox and the outbox folder

Refresh Time

in order to update the refresh time the admin user must follow this procedure

- disable the account and press save
- wait at least the “reloading time” (see InboxListener.xml). In this way the account related process that refresh the cache for the specific user should be stopped.
- update the refresh time , enable the account and press save.

After the last step the account related process that refresh the cache is started up

Soft-Delete on Server

If this option is enabled the syncml client can not send a delete command to the server.

For instance the user removes an item on the client inbox folder and press sync. The email is not deleted on the server.

In this way the user on your own device can implement the the scenario that allows to preserve some important emails on the device avoiding the delete on the server;

In order to create this environment on the device the user can:

- create an “archive” folder
- move the email from inbox to archive folder
- press sync

3.3.3. Configuring the Email Connector using the Command Line Tool

NOTE: Before using the Command Line Tool the administrator has to install the Inbox-Listener Tool. This command line tool is obsolete and the administrator should use the Funambol Administrator Console. This command line tool can be used just to investigate about the IMAP Folder for an account.

In order to create a new account for a Funambol DS-Server user the administrator must provide the following steps:

- Open a Shell
- go in the <installation dir>\bin directory
- type > ./inboxlistener.sh console

the main menu in the Figure 8 appears

```
INBOX LISTENER ACCOUNT CONSOLE

0      quit

11     list accounts
12     get account
13     insert account
14     delete account
15     disable account
16     enable account
17     update account
18     check account
19     get cached info

21     list public mail servers
22     get public mail server
23     insert public mail server
24     delete public mail server
25     update public mail server

31     IMAP Folder for account
```

Figure 8: Inbox Listener Main Menu

- command 11: list all the mailbox accounts that should be monitored
- command 12: show the account details
- command 13: create a new accounts

- command 14: delete an account
- command 15: disable a specific account
- command 16: enable a specific account
- command 17: update properties about a specific account
- command 18: check mail server access for a specific account
- command 19: list the email in the cache for a specific account
- command 21: list all the public mail server
- command 22: show the public mail server details
- command 23: create a new public mail server
- command 24: delete a public mail server
- command 25: update a public mail server
- command 31: show the imap folder list name for a specific account

This command line tool can be used just to investigate about the IMAP Folder for an account. (command 31)

- type 31
- type the username (the DS-Server username)

the result should be like this:

```
IMAP Folders in the Account gilberto.migliavacca
folder name: Deleted Items - full name: Deleted Items
folder name: Drafts - full name: Drafts
folder name: INBOX - full name: INBOX
folder name: Outbox - full name: Outbox
folder name: Sent Items - full name: Sent Items
```

Figure 9: Subscriber sample results

Note: this tool works only for IMAP-based Account. If in the administrator try to get information for a pop3-based Account the tool gets an error.

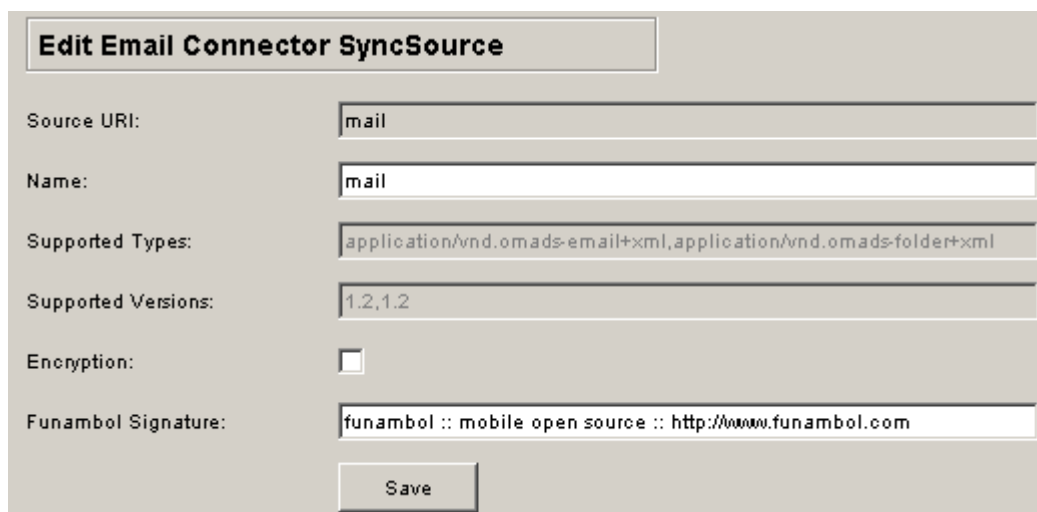
3.4. Configuring Email SyncSource

The email connector provides a single SyncSource type; all the users will be use the “mail” SyncSource in order to synchronize the emails



Figure 10: SyncSource access point

When the administrator user points on the Email SyncSource item he can see the Figure 11



Edit Email Connector SyncSource

Source URI: mail

Name: mail

Supported Types: application/vnd.omads-email+xml,application/vnd.omads-folder+xml

Supported Versions: 1.2,1.2

Encryption: ☐

Funambol Signature: funambol :: mobile open source :: http://www.funambol.com

Save

Figure 11: Email SyncSource Configuration Panel

In the SyncSource Panel the administrator can set the following properties:

Property	Description
Source URI	The sync source URI [i.e. "mail"].
Name	The SyncSource name. [i.e. "mail"]
Supported Type	application/vnd.omads-email+xml,application/vnd.omads-folder+xml
version	1.2,1.2
encryption	the communication between syncml client and ds-server will be encrypted (DES and BASE64)
Funambol Signature	This string is attached in all the email sent by the Funambol email plug-in and client

3.5. Officer Configuration

In order to set the Officer for the Funambol Email Connector, you have to set the parameters in the file; at the moment you have to set no parameters :

<installation dir>\ds-server\config\funambol\server\security\EmailOfficer.xml

This is an example of the EmailOfficer.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<java version="1.4.0" class="java.beans.XMLDecoder">
  <object class="com.funambol.email.security.EmailOfficer">
  </object>
</java>
```

In the Admin tool you have to specify the correct Officer:

Officer : com/funambol/server/security/EmailOfficer.xml

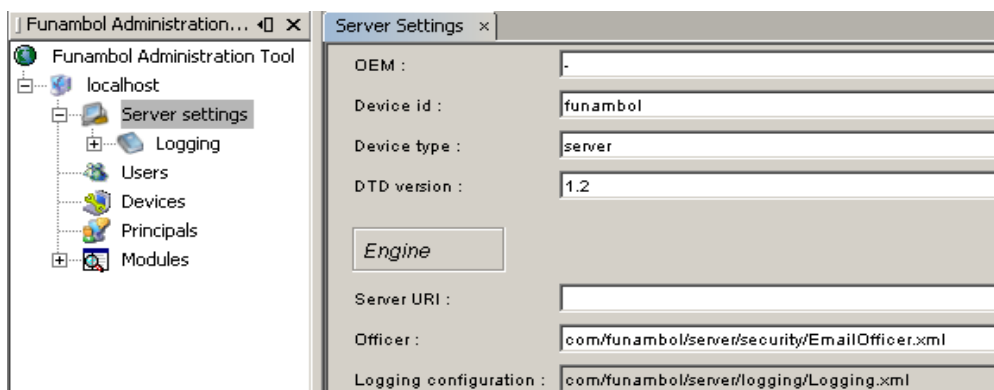


Figure 12: Officer Settings

The Main information about the EmailOfficer are in the Email Connector Configuration Panel. See next paragraph for details.

3.6. Encryption

In order to enable the Encryption Communication between syncML client (i.e. The Funambol WM plug-in) and the Funambol DS-Server/Email Connector you have to check the "encryption/encoding" checkbox in the SyncSource configuration Panel

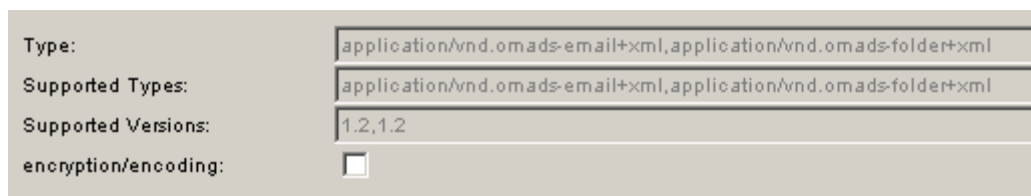


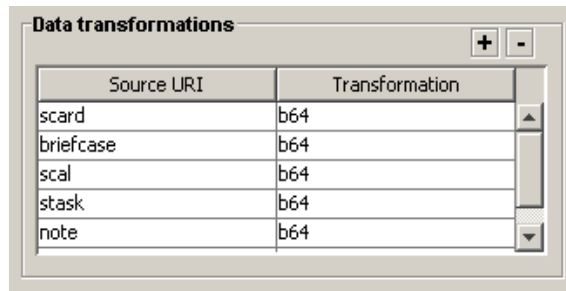
Figure 13: Encryption Settings

if the encryption is enabled the synchronization will be provided with DES and BASE64 encoding

You can also check the configuration in the Data transformation panel in the Server Settings section

- Open the Funambol Admin Tool
- Settings – Data Transformation – Configure

- In the DataTrasformer Configuration Manager; add a raw in the Data Transformations table



Source URI	Transformation
scard	b64
briefcase	b64
scal	b64
stask	b64
note	b64

Figure 14: Data Transfomation Table

- press “+”
 - SourceUri : set the SyncSource Name [i.e. “mail”]
 - Transformation: set “**des;b64**”
- press “save”

In this way the server is enabled to handle the encryption data. Meanwhile you have to set on the client the “encryption” property.

3.7. Enabling Log

3.7.1. Enabling log in the Email Connector

In order to configure the logging level for the Funambol Email Connector, you have to modify the file:

<installation dir>\ds-server\config\sync4j\server\logging\Logging.xml

adding this rows:

```
<void method="add">
  <object class="com.funambol.framework.config.LoggerConfiguration">
    <void property="append">
      <boolean>true</boolean>
    </void>
    <void property="count">
      <int>1</int>
    </void>
    <void property="inherit">
      <boolean>true</boolean>
    </void>
    <void property="level">
      <string>INFO</string>
    </void>
    <void property="limit">
      <int>100</int>
    </void>
    <void property="name">
      <string>funambol.email</string>
    </void>
    <void property="pattern">
      <string>logs/syncserver.email.log</string>
    </void>
  </object>
</void>
```

This file is re-written during server installation (or module installation) so, if you don't want to lose the changes, you may insert the same rows in the file:

```
<installation dir>\ds-server\default\config\common\beans\funambol\server\logging\Logging.xml
```

3.7.2. Enabling log in the InboxListener

In order to configure the logging level for the Funambol InboxListener, you have to modify the file:

<installation dir>\bin\log4j.xml

```
<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">

<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/">

  <appender name="console" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern"
        value="[%d{yyyy-MM-dd HH:mm:ss,SSS}] [%c] [%p] [%t] %m%n" />
    </layout>
  </appender>

  <appender name="log-file" class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="../log/il.log"/>
    <param name="MaxFileSize" value="100MB"/>
    <param name="MaxBackupIndex" value="5"/>
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern"
        value="[%d{yyyy-MM-dd HH:mm:ss,SSS}] [%c] [%p] %m%n" />
    </layout>
  </appender>

  <logger name="funambol.email">
    <level value="all"/>
  </logger>

  <logger name="org">
    <level value="error"/>
  </logger>

  <root>
    <appender-ref ref="console" />
    <appender-ref ref="log-file" />
  </root>

</log4j:configuration>
```

You can enable/disable the log on console put the following comment in the line

```
<!-- <appender-ref ref="console" /> -->
```

4. Appendix A - Notification System between Funambol DS-Server and Mobile Device

In this chapter we show briefly the main information about the communication between the Funambol DS-Server and the Mobile Device (Figure 15).

A detailed and updated description is included in the specific Funambol Server and Funambol Windows Mobile Plug-in documentation; this chapter must be consider as conceptual.

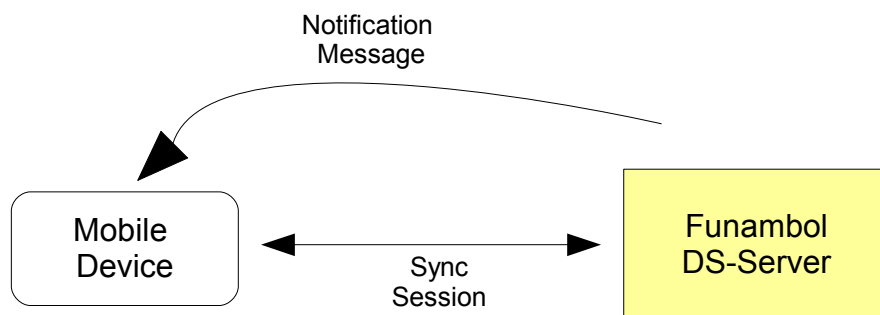


Figure 15: Communication between Server and Device

The Funambol DS-Server sends notification to the device using two protocol.

- **SMS protocol (WAP protocol):** the engine sends the notification sending an sms. In this case we need:
 - activate an SMS service (we have to ask a service to an SMS Provider)
 - provide a modules to install in the funambol engine (an s4j file ..)This module should implement the specification of the SMS provider (i.e. Send a http request with the sms to the provider server) In order to develop this module you can see the class `com/funambol/server/notification/sender/SimpleWAPSenderImpl.java` in the Objectweb cvs
- **TCP-IP protocol:** the engine send to the device a message using the tcp-ip protocol. In the case we need a Mobile Device with a public IP when the “push” is turned on.

When the client plug-in on the device receives the notification automatically performs an email synchronization via the Email Connector module.

Remember to set the push properties on the plug-in settings and to set the properties in the Admin tool

4.1. Mobile Device Settings

Note that this section includes the Funambol client configuration. The images could be out-of-date but the instruction will be pretty similar in the different Funambol client versions.

V6 platform; On the Mobile Device you have to set the following properties.

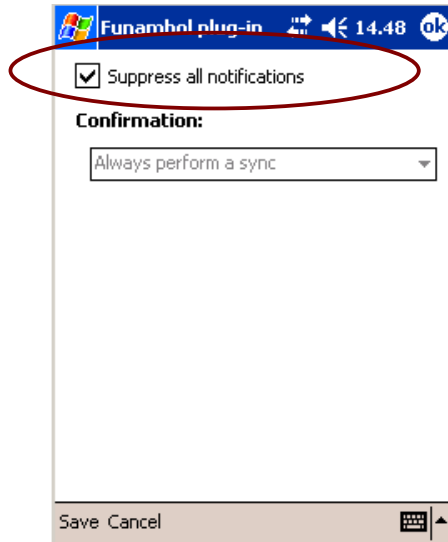


Figure 16: Mobile Device Settings

In order to disable/enable the notification the user must check/decheck the 'Suppress all notifications' option. If the user disables the option the plug-in enables the notification based on both wap and tcp-ip protocol, then the DS-Server will send the message with the protocol defined in the Admin Tool.

Previous Platform; On the Mobile Device you have to set the following properties.

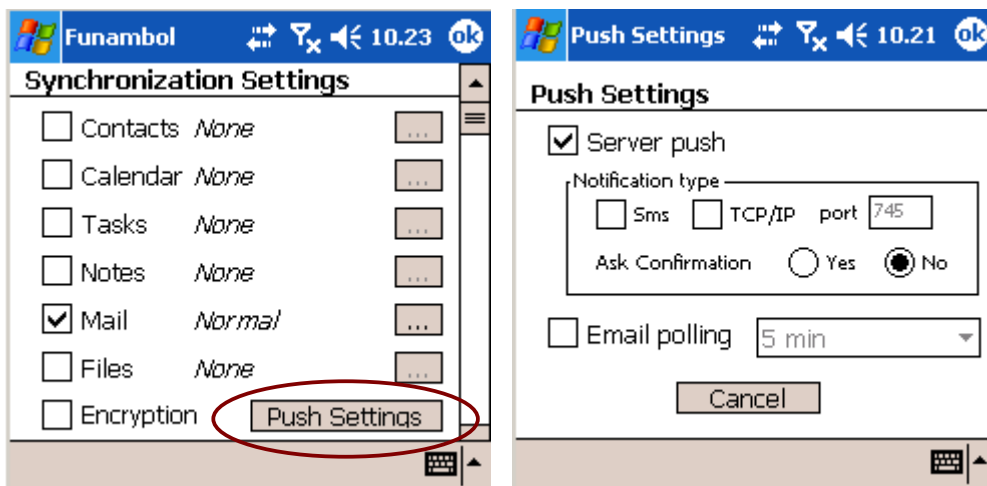


Figure 17: Mobile Device Settings

In the Push settings you can choose if the notification is via Sms or TCP/IP

When the user check “server push” and “tcp/ip” a message is sent to the Funambol DS-Server from the client in order to configure the “address” field in the admin tool (see next chapter).

When the user check “server push” and “sms” no messages are sent to the Funambol DS-Server from the device so the admin user must insert the “msisdn” field in the admin tool (see next chapter).

4.2. Admin Tool Settings

In order to set the properties for the DS-Server we have to use the Admin Tool.

- open the admin tool
- choose <server-name> ---> Devices ---> Edit

ID :	fwm-03E0C3804117A6822
Type :	
Timezone :	Not specified
	<input type="checkbox"/> Convert dates to this timezone
Charset :	UTF-8
Address :	
Phone Number :	
Notification Type :	OMA DS 1.2
Description :	NOKIA OMA DS 1.1 OMA DS 1.2

Figure 18: DS - Server Settings

We have to handle the following properties

- **Address:** IP address of the device (if the user chose the tcp/ip notification).
- **Msisdn:** Msisdn of the device (i.e., the phone number, if the user chose the sms notification)
- **Notification Type:**
 - empty
 - nokia oma ds 1.1: (not used for the email environment)
 - **oma ds 1.2:** this is the default **tcp/ip notification mode** for the email environment. The server uses a **builder component** to create notification messages and a **sender component** to send notification messages to the device. The **builder component** is the DSNotificationBuilder.xml already present in the previous version; the **sender component** is the PushSender.xml (ex - TCPIPSender.xml present in the previous version).

The default PushSender.xml contains the tcp/ip mode configuration:

```
<?xml version="1.0" encoding="UTF-8"?>
<java version="1.4.2_01" class="java.beans.XMLDecoder">
  <object class="com.funambol.server.notification.sender.TCPIPSender">

    <!-- Connection timeout in seconds -->
    <void property="timeout">
      <int>60</int>
    </void>

    <!-- Numbers of attempts -->
    <void property="attempts">
      <int>3</int>
    </void>

    <!-- Waiting time between the attempts in seconds -->
    <void property="waitingTime">
      <int>20</int>
    </void>
```



```
<!-- Default port -->  
<void property="defaultPort">  
  <int>745</int>  
</void>  
  
</object>  
</java>
```

If the Administrator wants to implement the **SMS notification mode**, he must create and install a new module (a custom push sender s4j module). Note that Funambol provides an sms push sender module for the Carrier Edition (the funambol push sender s4j module)

The Custom Push Sender

The administrator should implements an s4j module and he can follow the main indication in the sample class: `com/funambol/server/notification/sender/SimpleWAPSenderImpl.java` in the Objectweb cvs.

The implementation can contains a PushSender.xml with the http configuration. This configuration on the SMS Provider specification.

5. Appendix B – Mail Server Configuration

5.1. Microsoft Exchange 2000/2003

We can use imap-based account or pop3-based account

We have to setup the MS Exchange Server

- creation of the user in the MS Exchange
- login using “outlook web access” or an account imap on outlook express / outlook

SyncSource Folder Name Configuration

Folder name

- inbox: inbox
- outbox: outbox
- sent: Sent Items
- drafts: Drafts
- trash: Deleted Items

5.2. Domino 6.5/7.0

About imap-based account we have to consider that some aspects of a mail file are structured in template items that are visible only to a Notes client, and as such are not available to IMAP clients.

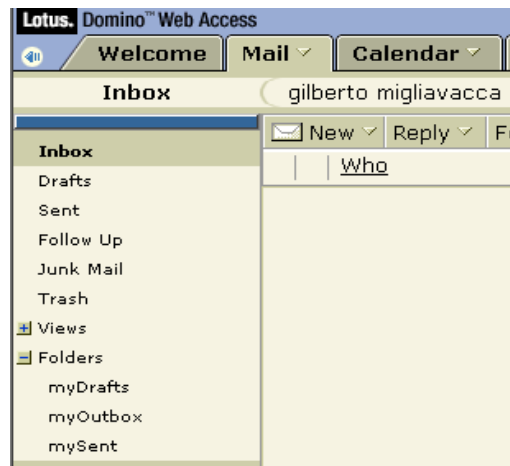
As a result, IMAP clients display certain folders and views in a mail file differently from Notes clients. For instance, from an IMAP client, the Inbox and Trash folders, and any public folders, appear as IMAP mailboxes.

Also, hidden and private folders are not visible to IMAP clients. And finally, IMAP clients do not display views that are part of the Notes mail file template, such as the Draft and Sent view

We have to setup the “domino” mail server

- creation of the user
- login using “Domino Web Access”
 - create the public folder in “folder”
 - pubOutbox
 - pubSent
 - pubDrafts

Note: the “sent” and “draft” folders are “private folder”. The Email connector will use the “public folder”. As a result, the connector displays/uses some folder/email in a mail file differently from Notes clients.



SyncSource Folder Name Configuration

Folder name

- inbox: inbox
- outbox: myOutbox
- sent: mySent
- drafts: myDrafts
- trash: Trash

5.3. GMail

We can use a pop3-based account

In order to config the account you have to set the following parameters

<i>field</i>	<i>value</i>
Incoming server	pop.gmail.com
Incoming server Port	995
Outgoing server	Smtp.gmail.com
Outgoing server Port	465
SSL	checked
Keystore path	See next info
Keystore password	See next info

The Gmail pop access need SSL configuration.

1) Download the GMAIL certificate from link:

https://www.geotrust.com/resources/root_certificates/certificates/Equifax_Secure_Certificate_Authority.cer

or

using cygwin type the command

wget

https://www.geotrust.com/resources/root_certificates/certificates/Equifax_Secure_Certificate_Authority.cer

2) Use the java tool in order to create the keystore (using the certificate just downloaded). open a shell (i.e. A dos shell) and type the command

```
> keytool -genkey -alias gmailcert -keyalg RSA -keystore gmailcertkeys
```

Note: you have to install the jdk on the machine

3) Import the certificate in the keystore:

```
> keytool -import -file Equifax_Secure_Certificate_Authority.cer -keystore gmailcertkeys
```

4) Set the syncsource and the connector to use the SSL parameter

5.4. Yahoo

We can use a pop3-based account

We have to setup the “yahoo.com” mail server.

- creation of the user
- login using “web access”

SyncSource Configuration

Source URI:	pyahoo				
Name:	pyahoo				
Outgoing Server :	smtp.mail.yahoo.it	Port:	587	Auth:	<input checked="" type="checkbox"/>
Incoming Server :	pop.mail.yahoo.it	Port:	110		

Note: yahoo server needs the authentication in order to send the email

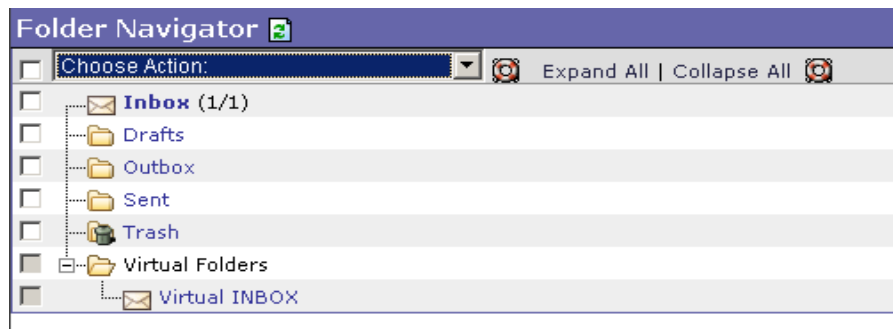
Note: currently the yahoo server needs the “Yahoo Mail Plus” option (it's not a free option) in order to allow the POP configuration.

5.5. Courier (Horde Web Access)

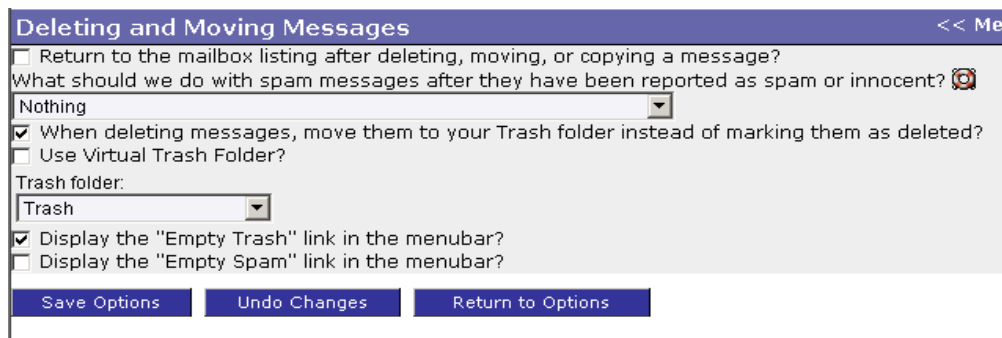
We can use: imap-based account and pop3-based account

We have to setup the “mail.*****.com” mail server (Courier Imap Server).

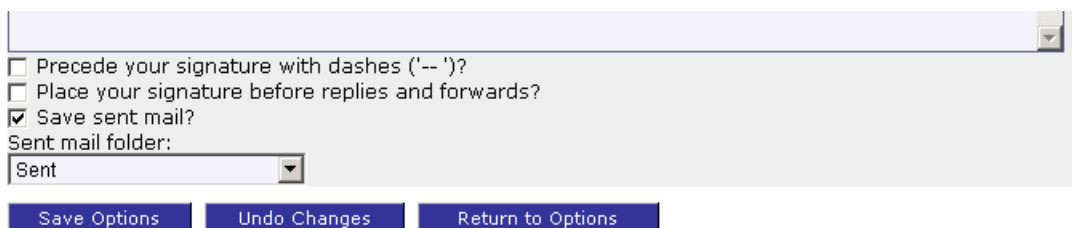
- creation of the user
- login using “Horde Web Access”
 - select INBOX on the left panel
 - select “folder” in the top bar
 - choose action: create folder
 - create: Sent, Outbox, Drafts



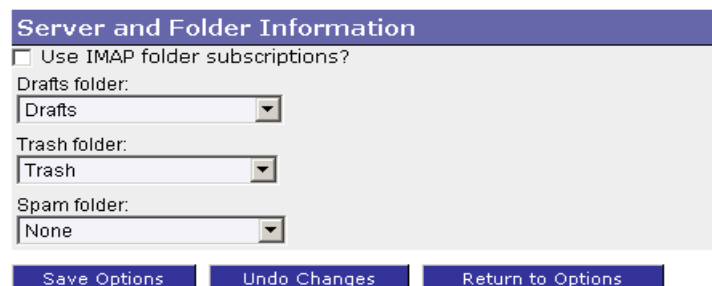
- select “options”
- choose “deleting and moving messages” box
- select the following option



- select “options”
- choose “Personal Information” box
- select the following option (at the end of the panel)



- select “options”
- choose “Server and Folder Information” box
- select the following option



SyncSource Folder Name Configuration

Folder name

- inbox: Inbox
- outbox: INBOX.Outbox
- sent: INBOX.Sent

- drafts: INBOX.Drafts
- trash: INBOX.Trash

5.6. CommuniGate Server 5.x

We tested the server with the pop3-based account. The configuration is standard but in the CommuniGate Server we have to specify as inbox folder name the value: 'INBOX' (all capital letters)

At the moment to do this operation the admin must insert the value in the directly in the DB schema:

- `update fnbl_email_mailserver set inbox_name='INBOX' where mailserver_id=????`
- `update fnbl_email_account set inbox_name='INBOX' where mailserver_id=????`