

ESMEMaillet

dakoller@googlemail.com - <http://twitter.com/dakoller>

2008-12-25

What is it and how can I use it?

ESME (to be found at <http://incubator.apache.org/esme/>) stands for Enterprise Social Messaging Experiment and was recently accepted by the Apache Incubator as a future Apache project. It provides an end user and process related microblogging infrastructure, which is - in contrast to Twitter, Yammer etc. - useable in the IT premises of a company, if it is not acceptable to use publicly accessible blogging infrastructure.

The ESME Maillet is a plugin for the [Open source Mail Server Apache James](#), which allows users of an ESME infrastructure (both humans and technical systems) to send email messages to an ESME instance using their own identity. Messages sent through the maillet show up as personal messages from these end users, so people following you get them in the respective blogging timeline.

How does it work?

- ESME provides a REST-like API, which is bundled in <http://code.google.com/p/esmeproject/source/browse/trunk/esme-nwlogger/lib/esme-rest-api-0.1-SNAPSHOT.jar> .
- An String-formatted token is needed to login to ESME impersonating a certain user. An end user can create a token on the ESME web interface at "Manage tokens".
- A database table holds the information for which email sender which token is to be used: every end user can register and unregister a token with an email to the mail server. An existing token is deleted in that case. If the database table does not exist, it will be created as a part of the first James-run with the maillet installed.
- The subject of an incoming email message is posted to ESME.
- Sending message to ESME, Registering an unregistering of token is confirmed by the mail server with a reply message.

Installation

The italic text is to be customized to your local environment.

x:\ stands here for the drive/directory, where you choose to install James.
aa.bb.com stands for the hostname, where the mail server can be reached.¹

Prerequisites

The installation is described in Windows terms: however no platform specific steps are needed.

1. You may also use a plain IP address, but be aware that in an email address the IP address has to be surrounded by []. (such as `postmaster@[192.168.2.1]`)

Even if you have already mail servers in your infrastructure, Apache James fits through its integrated Fetchmail: James can be instructed to use mailboxes, which already exist. For this tutorial I used version 2.3.1 of Apache James, which was the latest stable version.

You need access to a writeable, JDBC-compliant database: I used MySQL in this example.

Installation of the Apache James Mail Server

1. Apache James can be downloaded from <http://james.apache.org/download.cgi>.
2. Please refer to the installation instructions at http://james.apache.org/server/2.3.1/installation_instructions.html .

Installation of the Mailet

1. **Modify config.xml (usually to be found at x:\apps\james\SAR-INF): include the following line**

```
<mailetpackage>us.esme.mailet.ESMEMailet</mailetpackage>
```

as a child element of the `<mailetpackages>`-element.

2. **Modify the same file: include the following text**

```
<mailet match="RecipientIs=mail-me@aa.bb.com,register-  
token@aa.bb.com,unregister-token@aa.bb.com"  
class="us.esme.mailet.ESMEMailet">  
    <api_url>http://esmecloudserver.dickhirsch.staxapps.net/  
api</api_url>  
    <driver>com.mysql.jdbc.Driver</driver>  
    <dburl>jdbc:mysql://localhost:3306/esme_mailet</dburl>  
    <user>esme</user>  
    <password>master</password>  
    <max>20</max>  
    <table_prefix>esme_</table_prefix>  
    <esme_host>aa.bb.com</esme_host>  
</mailet>
```

as a child element of the `<processor name="root">`-Element.

The first line contains the email addresses for later use: the part before the hostname is hardcoded in the mailet. The hostname part does not matter.

You actually do not have to have user accounts for mail-me@...: it is just easier to follow, what happens inside, if you create these as local mailboxes. This might be needed for auditing reasons.

3. **Put the needed java libraries to X:\lib.**

The needed libraries are:

- ESMEMailet.jar (currently residing at www.dkoller.de/ESMEMailet.jar),
- Apache Commons Logging (used version 1.1.1 at http://commons.apache.org/downloads/download_logging.cgi),

- Apache Commons Codec (used version 1.3 at http://commons.apache.org/downloads/download_codec.cgi),
 - Apache Commons HttpClient (used version 3.2 at <http://hc.apache.org/downloads.cgi>),
 - ESMERestAPI (wrapper jar file, to be retrieved here: <http://code.google.com/p/esmeproject/source/browse/trunk/esme-nwlogger/lib/esme-rest-api-0.1-SNAPSHOT.jar>)
- and
- the necessary JDBC driver Jar files.

Other (in my case less useful) methods to deploy the maillet are described at http://james.apache.org/server/2.3.1/custom_maillet.html).

4. Instruct Fetchmail to retrieve mails from already existing mailboxes and to forward them to the defined addresses (Optional).

The fetchmail configuration is at x:\apps\james\conf\james-fetchmail.xml.

Set fetchmail-enabled to "true" and create a new <fetch>-Element under the <fetchmail>-Element with the following content:

```
<fetchmail enabled="true">
  <fetch name="remote_mail_domain">
    <accounts>
      <!-- <alllocal userprefix="" usersuffix="@myaccount"
password="password" recipientprefix="" recipientsuffix="@mydomain.com"
ignorercpt-header="true"/> -->
      <account user="remote-username" password="remote-password"
recipient="mail-me@aa.bb.com" ignorercpt-header="true"/>
      <account user="remote-username" password="remote-password"
recipient="register-token@aa.bb.com" ignorercpt-header="true"/>
      <account user="remote-username" password="remote-password"
recipient="unregister-token@aa.bb.com" ignorercpt-header="true"/>
      <!-- Custom recipient header handling -->
      <!-- this example search an header named X-Envelope-To and
use the recipient found in that header -->
      <!--
      <account customrcpt-header="X-Envelope-To"
user="myglobalaccount" password="password" recipient="postmaster@localhost"
ignorercpt-header="false"/>
      -->
    </accounts>
    <!-- The address of the server from which to fetch the mail -->
    <host>hostname_of_mailserver</host>
    <!-- How frequently this host is checked - in milliseconds.
600000 is every ten minutes -->
    <interval>60000</interval>
  </fetch>
</fetchmail>
```

```
<!-- name of the javamail provider you wish to use -->
<!-- (pop3, imap, etc. -->
<javaMailProviderName>pop3</javaMailProviderName>

<!-- name of the folder to open -->
<javaMailFolderName>INBOX</javaMailFolderName>

<!-- Properties to be applied to the JavaMail Session. -->
<!-- Properties are specific to the selected JavaMail provider. -->
->
<!-- Any number may be specified. -->
<javaMailProperties>
  <!-- Set the connection timeout to 3 minutes -->
  <property name="mail.pop3.connectiontimeout" value="180000"/>
  <!-- Set the I/O timeout to 3 minutes -->
  <property name="mail.pop3.timeout"
value="180000"/>
  </javaMailProperties>

<fetchall>>false</fetchall>

<recursesubfolders>>false</recursesubfolders>

<fetched leaveonserver="false" markseen="true"/>

<remotereceivedheader index="1" reject="true"
leaveonserver="true" markseen="false"/>

<maxmessagesize limit="0" reject="false" leaveonserver="true"
markseen="false"/>

<undeliverable leaveonserver="true" markseen="false"/>

<recipientnotfound defer="true" reject="true"
leaveonserver="true"
markseen="false"/>

<blacklist reject="true" leaveonserver="true" markseen="false">
wibble@localhost, flooble@localhost </blacklist>

<userundefined reject="true" leaveonserver="true"
markseen="false" />

<remoterecipient reject="true" leaveonserver="true"
markseen="false" />
</fetch>
</fetchmail>
```

5. Start the mail server with x:\bin\run.bat.

Now users can send their ESME token as subject to register-token@aa.bb.com: his/her token delivery gets confirmed.

After that he/she can send messages to mail-me@aa.bb.com, the subject of the message gets immediately sent to ESME.

In case a token was compromised, a user can change it sending a new token to register-token@aa.bb.com.

A user may also decide to unregister his token with a message to unregister-token@aa.bb.com.

Operations

After installing the mailet the status of the mailet can be seen in the respective log files:

- X:\apps\james\logs\mailet-<date>.log (for the mailet operation) and
- X:\apps\james\logs\fetchmail-<date>.log (for the fetchmail operation).

Open issues

- The text of the email and existing attachments are currently ignored: a mail text and attachment storage service (with an included URL shortener) could be an option here.
- If ESME cannot be reached when the mailet processes the mail, the message cannot be published: it is also not saved for later processing.
- Every ESME message out of the Mailet is tagged with "ESMEMailet": it is currently not possible for the end user to customize the tag. (any proposals how a handling for that could look like?)

Please give me feedback, when something does not work as expected or if you have comments and improvement proposals.