ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-net

Peter Reutemann

February 8, 2012

DRAFT

©2012

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Contents

1 Email **7**

2 FTP **9**

3 SSH **11**

4 Miscellaneous **13**

Bibliography **15**

List of Figures

1.1 Email setup dialog 8

Chapter 1

Email

Flows are ideal for being run as background jobs ("-headless" flag). For example, importing or processing data in batches can be done at night time. Of course, you want to be notified if something went wrong or some predictions are off. Adding the *Email* sink to existing flows, allows for automatic sending of emails: if everything is OK then send an email to the customer, otherwise send an email to sysadmin. The *Email* actor adds all incoming file names, e.g., the array output of a *DirectoryLister* source, as attachments before sending the email off to the specified recipiqents. You can also define a custom subject and body.

In order to be able to send emails, ADAMS needs to know what SMTP server to connect to. The following example configures ADAMS to send emails using a Gmail account [1].

The easiest way of configuring Email, is to use the dialog available from the main menu (*Program* → *Email setup*), as depicted in Figure 1.1. The placeholders *USER* and *PASSWORD* have to be replaced with the actual user credentials and *YOUR NAME* with the actual user's name, of course.

Alternatively, your can also simply create a properties file in the `$HOME/.adams` directory, called `Email.props`. The content for the Gmail setup would look like this:

```
Enabled=true
SmtpStartTls=true
SmtpPassword=
SmtpServer=smtp.gmail.com
SmtpPort=587
SmtpRequiresAuthentication=true
SmtpUser=USER
DefaultAddressFrom=YOUR NAME <USER@gmail.com>
DefaultSignature=
```

[1] For more details, see the following Gmail help page:
`http://support.google.com/mail/bin/answer.py?hl=en&answer=13287`

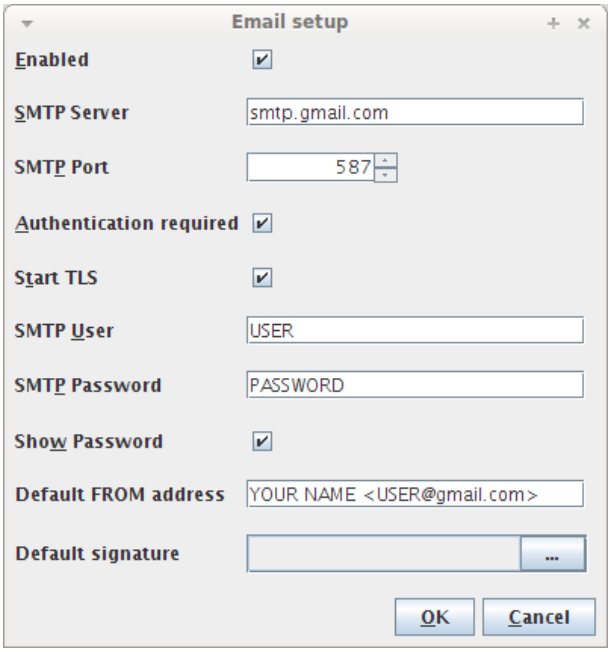

Figure 1.1: Email setup dialog

Chapter 2

FTP

In addition to Email, ADAMS also supports FTP (file transfer protocol). Since FTP does not encrypt the commands or transferred data, it is recommended to not use it outside a company's network or for non-anonymous access (e.g., download from a public FTP server).

The following actors are available:

- *FTPConnection* – standalone actor that defines the server connection.
- *FTPLister* – lists the files and directories on a remote server.
- *FTPGet* – for obtaining a remote file via FTP.
- *FTPSend* – for sending a file via FTP to a remote host.

Chapter 3

SSH

ADAMS also comes with SSH support, thanks to the JSch library [1]. This allows for secure remote access. At the time of writing, RSA [2] and DSA [3] are supported for public key encryption schemes.

The following actors are available:

- *SSHConnection* – standalone actor that defines the server connection.
- *SSHExec* – for executing remote commands.
- *ScpFrom* – for copying a file *from* a remote host using secure copy.
- *ScpTo* – for copying a file *to* a remote host using secure copy.
- *SFTPGet* – for obtaining a remote file via secure FTP.
- *SFTPSend* – for sending a file via secure FTP to a remote host.

[1] `http://www.jcraft.com/jsch/`
[2] `http://en.wikipedia.org/wiki/RSA_%28algorithm%29`
[3] `http://en.wikipedia.org/wiki/Digital_Signature_Algorithm`

Chapter 4

Miscellaneous

Some other basic, but useful actors are the following:

- *Browser* – opens the system's default browser with the specified URL.
- *DownloadFile* – downloads a file via HTTP.

Bibliography

[1] *ADAMS* – Advanced Data mining and Machine learning System
`http://adams.cms.waikato.ac.nz/`

[2] *Gmail* – Configuring other mail clients
`http://support.google.com/mail/bin/answer.py?hl=en&answer=13287`