JAVAMAIL API - BOUNCED MESSAGES

http://www.tutorialspoint.com/javamail_api/javamail_api_bounced_messages.htm

A message can be bounced for several reasons. This problem is discussed in depth at rfc1211. Only a server can determine the existence of a particular mailbox or user name. When the server detects an error, it will return a message indicating the reason for the failure to the sender of the original message.

There are many Internet standards covering Delivery Status Notifications but a large number of servers don't support these new standards, instead using ad hoc techniques for returning such failure messages. Hence it get very difficult to correlate the *bounced* message with the original message that caused the problem.

JavaMail includes support for parsing Delivery Status Notifications. There are a number of techniques and heuristics for dealing with this problem. One of the techniques being Variable Envelope Return Paths. You can set the return path in the enveloper as shown in the example below. This is the address where bounce mails are sent to. You may want to set this to a generic address, different than the From: header, so you can process remote bounces. This done by setting *mail.smtp.from* property in JavaMail.

Create Java Class

Create a java class file **SendEmail**, the contents of which are as follows:

```
import java.util.Properties;

import javax.mail.Message;
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;

public class SendEmail {
   public static void main(String[] args) throws Exception {
      String smtpServer = "smtp.gmail.com";
      int port = 587;
      final String userid = "youraddress";//change accordingly
      final String password = "******";//change accordingly
      String contentType = "text/html";
      String subject = "test: bounce an email to a different address " +
    "from the sender";
      String from = "youraddress@gmail.com";
      String to = "bouncer@fauxmail.com";//some invalid address
      String bounceAddr = "toaddress@gmail.com";//change accordingly
      String body = "Test: get message to bounce to a separate email address";

      Properties props = new Properties();

      props.put("mail.smtp.auth", "true");
      props.put("mail.smtp.starttls.enable", "true");
      props.put("mail.smtp.host", smtpServer);
      props.put("mail.smtp.port", "587");
      props.put("mail.transport.protocol", "smtp");
      props.put("mail.smtp.from", bounceAddr);

      Session mailSession = Session.getInstance(props,
         new javax.mail.Authenticator() {
            protected PasswordAuthentication getPasswordAuthentication() {
               return new PasswordAuthentication(userid, password);
            }
         });

      MimeMessage message = new MimeMessage(mailSession);
      message.addFrom(InternetAddress.parse(from));
      message.setRecipients(Message.RecipientType.TO, to);
      message.setSubject(subject);
      message.setContent(body, contentType);

      Transport transport = mailSession.getTransport();
      try {
         System.out.println("Sending ....");
         transport.connect(smtpServer, port, userid, password);
         transport.sendMessage(message,
            message.getRecipients(Message.RecipientType.TO));
         System.out.println("Sending done ...");
      } catch (Exception e) {
         System.err.println("Error Sending: ");
         e.printStackTrace();

      }
      transport.close();
   }// end function main()
}
```

Here we can see that the property *mail.smtp.from* is set different from the *from* address.

Compile and Run

Now that our class is ready, let us compile the above class. I've saved the class SendEmail.java to directory : **/home/manisha/JavaMailAPIExercise**. We would need the jars *javax.mail.jar* and *activation.jar* in the classpath. Execute the command below to compile the class $boththejarsareplacedin/home/manisha/directory$ from command prompt:

```
javac -cp /home/manisha/activation.jar:/home/manisha/javax.mail.jar: SendEmail.java
```

Now that the class is compiled, execute the below command to run:

```
java -cp /home/manisha/activation.jar:/home/manisha/javax.mail.jar: SendEmail
```

Verify Output

You should see the following message on the command console:

```
Sending ....
Sending done ...
```

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js