

Technical Explorations

by Keith R. Bennett

Mailing Files Programmatically with GMail

Jul 25, 2012 • keithrbennett

Recently I had the need to send someone multiple 1-2 MB files. Email was the best option, for reasons that are not very interesting, and outside the scope of this article.

I naturally turned to my favorite language, Ruby. I did a little research into the various gems and their configuration, and offer this simple example (source code below) in the hope that it will save you a little time if you have the same need.

I've tried to trim out anything not relating to the actual sending of the mail so that you can more easily understand it and adapt it for your use.

- Keith

```
1  #!/usr/bin/env ruby
2  # Mails a file using GMail's SMTP Server.
3  # For illustrative purposes; error checking and testing intentionally omitted for brevity.
4  #
5  # Requirements:
6  #   1) the 'mail' gem must be installed
7  #   2) a file named 'pw.txt' containing the Google password must be present
8  #      in the current directory.
9  #
10 # Ruby versions: tested on 1.9.3, 1.8.7, JRuby
11 #
12 # Keith Bennett
13
14 require 'rubygems' # Need this for Ruby 1.8
15 require 'mail'
16
17
18 ATTACHMENT_FILESPEC = 'attachment.txt'
19 File.open(ATTACHMENT_FILESPEC, 'w') do |file|
20   file << "Sample attachment text file, sent #{Time.now}\n"
21 end
22
23
24 # Used for GMail authentication and the 'from' header field.
25 def my_gmail_address
26   # Return your gmail address, e.g.:
27   'me@gmail.com'
28 end
```

```
29
30
31 # Used for 'to' header field
32 def recipients
33   # a single address or a comma separated list of email addresses, e.g.:
34   'mom@aol.com,fred@people.com'
35 end
36
37
38 # GMail password for SMTP authentication.
39 # Create a file named 'pw.txt' and put your password there in plain text.
40 # Not the best way to do this!
41 def password
42   File.read('pw.txt')
43 end
44
45
46 def body_text
47   "This message contains the following attachment: #{ATTACHMENT_FILESPEC}.\nSent #{Time.now}
48 end
49
50
51 def subject_text
52   "Gmail-It Attached File: #{ATTACHMENT_FILESPEC}"
53 end
54
55
56 # This information could probably be put in the Mail.deliver block below,
57 # but it's here in case multiple mail calls are made.
58 Mail.defaults do
59   delivery_method :smtp, {
60     :address      => "smtp.gmail.com",
61     :port         => 587,
62     :domain       => 'gmail.com',
63     :user_name    => my_gmail_address,
64     :password     => password,
65     :enable_starttls_auto => true  }
66 end
67
68 puts "Delivering file: #{ATTACHMENT_FILESPEC} to #{recipients}."
69
70 Mail.deliver do
71   from      my_gmail_address
72   to        recipients
73   subject   subject_text
74   body      body_text
75   add_file  ATTACHMENT_FILESPEC
```

```
76  end
77
78  puts "File #{ATTACHMENT_FILESPEC} sent."
79  puts "Check recipient email account(s) (#{recipients}) to verify success."
```

mail-using-gmail.rb hosted with ❤ by GitHub

[view raw](#)

Published with [GitHub Pages](#)