

Mailing CSV files from PostgreSQL



On TERENA systems, this script is located at `/usr/local/share/sysadmin/misc/pgmailcsv`.

We often end up creating CSV files from the results of some specific PostgreSQL query, and mailing that around. This usually involves starting up a GUI database tool (we use EMS PostgreSQL manager), clicking through some windows, then saving a CSV file, then mailing it.

Or manually doing everything from a shell. But wouldn't it be nice if this could all be done in one go? The following shell script takes care of that:

```
#!/bin/bash
#
# Script to e-mail the output of an SQL query
# Dick Visser <visser@terena.org> 20120214
# $Id: pgmailcsv 192 2012-02-17 10:18:27Z visser $
if [ $# -ne 3 ]; then
    echo "Error..."
    This script needs exactly THREE arguments:
        1. Database name
        2. E-mail address
        3. SQL query, surrounded by DOUBLE QUOTES
    Example:
    $0 tnc2012 webmaster@terena.org \"SELECT fname FROM users\"
    Long or complicated queries can be stored in an SQL text file:
    $0 tnc2012 webmaster@terena.org \"\`cat complicated.sql\`\"
    "
    exit
else
    OUTFILE="/tmp/PostgreSQL_Output_`date +%F_%Hh%Mm%Ss`.csv"
    touch "$OUTFILE"
    # LATIN1 is needed, UTF8 borks out on CSV
    psql "$1" -Atc "COPY ($3) to '$OUTFILE' WITH CSV HEADER"
    echo -en "\nPlease find attached the CSV formatted results of this SQL query to the '$1' database:\n
    $3\n" | mutt -a "$OUTFILE" -s "PostgreSQL query results from '$1'" -- "$2"
    rm "$OUTFILE"
fi
```

For this work, you need enough privileges, so `su` to user `postgres` before running this.

Also, you need a working `mutt` installed.

In case it wasn't clear, this script is meant to *view* data, which implies `SELECT`. So don't do stupid things like `UPDATE` or `DELETE`.

CSV files with e-mail addresses

The generated CSV files can be used by the Mailtweak Thunderbird plugin, so that a mail merge can be done by end users.

Unfortunately this process is a bit tricky when names include UTF-8 characters:

```
mydatabase=# SELECT fullname || ' <' || email || '>' AS email FROM users WHERE email = 'horvath@terena.org';
          email
-----
 Gyöngyi Horváth <horvath@terena.org>
(1 row)
```

This is a problem with MailTweak, because it doesn't grok MIME encoding, so it will use this string verbatim in the mail header, which which screw up the text (and likely trip any upstream spam filters).

This can be fixed by using Perl's [Encode::MIME::Header](#). I choose to do this from within PostgreSQL, by adding the Perl procedural language. I needed to use the "unsafe" version, so that libraries can be used.

To do this:

- Install the PL/Perl procedural language for PostgreSQL: `apt-get -s install postgresql-plperl-8.4`
- Add it to your database. You need `postgres` superuser privileges to do so:

```
psql mydatabase -Atc "CREATE LANGUAGE plperl"
```

- Add the following function to your database:

```
CREATE OR REPLACE FUNCTION public.mime_encode (  
    text  
)  
RETURNS text AS  
$body$  
use Encode qw/encode/;  
return encode('MIME-Header', $_[0]);  
$body$  
LANGUAGE 'plperl'
```

At this point you can use your new function to generate nicely formatted MIME headers:

```
mydatabase=# SELECT mime_encode(fullname) || ' <' || email || '>' AS email FROM users WHERE email =  
'horvath@terena.org';  
  
-----  
=?UTF-8?B?R3nDtm5neWkgSG9ydsOhdGg=?= <horvath@terena.org>  
(1 row)
```

MailTweak and Thunderbird now can digest the addresses, and UTF-8 will work OK 😊