

mepoadm / August 09, 2013 11:30PM

[Backup and Transfer a LAMP\(LAPP\) System: Linux/Apache2/PostgreSQL/PHP](#)

1. [1. User accounts](#)
2. [2. User Files \(/home\)](#)
3. [3. Database \(PostgreSQL\)](#)
4. [4. Web Server \(Apache\)](#)
5. [5. PHP](#)
6. [6. crontab](#)
7. [7. References](#)

Assume we have an installed Linux system on hand. We have six parts to transfer: (1) User accounts; (2) User files (/home); (3) Database; (4) Apache; (5) PHP; and (6) crontab.

1. User accounts

Copy and transfer the following user-related files and directories:

. /etc/passwd - File - contains various pieces of information for each user account . /etc/shadow - File - contains the encrypted password (and aging) information for user's accounts. . /etc/group - File - defines the groups to which users belong . /etc/gshadow - group shadow file (contains the encrypted password for group)
. /var/spool/mail - Generally user emails are stored here. . /home - All Users data is stored here.
Do remember to "preserve (-p)" file info (permissions, owner, etc) in the transfer. Do not copy-and-paste the passwd/shadow files directly. It's recommended to only attach the human-added users into the new system user account files.

2. User Files (/home)

(1) rsync

Transfer a certain user folder:

```
rsync -av /home/user/ account@domain:/home/user/
```

Or

(2) Tar

cd /

```
tar -cvpf /archive_folter/home-YYYYMMDD.tar home --exclude=home/exclude_foler
```

Untar:

cd /destination_folder

```
tar -xvpf home-YYYYMMDD.tar
```

3. Database (PostgreSQL)

Backup (export):

```
Shell> pg_dumpall -U postgres | bzip2 > cluster_YYYYMMDD.bz2
```

Transfer (scp) and unzip (bzip2 -d) the database cluster file.

Restore (import):

```
Shell> psql -f cluster_YYYYMMDD -U postgres postgres
```

Config:

shared_buffers = 2GB (1/4 of the memory is a reasonable starting value)

work_mem = 25MB

effective_cache_size = 4GB (1/2 of total memory would be a normal setting)

Password File:

~/.pgpass

4. Web Server (Apache)

(1) Configuration:

/etc/apache2/*.conf

/etc/apache2/vhost.conf/*.conf

(2) Modules:

vi /etc/sysconfig/apache

Add APACHE_MODULES: php5, headers, rewrite

Set APACHE_SERVER_FLAGS="SSL"

(3) Remove manual directory:

```
mv /etc/apache2/conf.d/manual.conf /etc/apache2/conf.d/manual.conf.ori
```

5. PHP

1. Install the modules:

```
zypper install php7-curl php7-gb php7-mbstring php-openssl php7-pgsql
```

2. Edit php.ini (/etc/php5/cli/php.ini)

short_open_tag = On

output_handler = ob_gzhandler

memory_limit = 512M

upload_max_filesize = 8M

date.timezone = 'Asia/Taipei'

session.cookie_secure = 1

session.cookie_httponly = 1

6. crontab

(1) Edit crontab

crontab -e
(2) Add jobs

```
0 0 * * * pg_dump -U postgres database > database_$(date +%Y%m%d%H).sql
```

7. References

User Accounts:

<http://www.cyberciti.biz/faq/howto-move-migrate-user-accounts-old-to-new-server/>

<http://serverfault.com/questions/19274/how-to-transfer-user-accounts-to-a-new-linux-machine>

Disk Partitioning:

<https://help.ubuntu.com/community/DiskSpace>

Database:

https://wiki.postgresql.org/wiki/Tuning_Your_PostgreSQL_Server

Edited 18 time(s). Last edit at 02/28/2019 10:55PM by mepoadm.
