

Install a Drupal Customer Portal

Created by Paul Chapotet, last modified by Rohan Singh on Aug 10, 2016

- [Requirements and informations](#)
- [Installation of PHP](#)
- [Installation of Apache, Mysql and Drupal](#)
- [Copy the data from another portal \(Optional\)](#)
- [Install lpms and ccpark services](#)
- [Set up scripts](#)
- [Set up domains and Reverse Proxy](#)
- [Finish your installation](#)
- [Automate import of sqlite database from production environments \(ideal for testing environments\)](#)
- [Clean automatically your LPMS/CCPARK/Drupal environment](#)

Requirements and informations

Important information

- This installation has been tested on Ubuntu 12.04.x LTS
- ANAI D6 modules should be in /sites/all/modules/d6/, they are available on github: <https://github.com/QPcorp/d6>
- We use a theme named "metropolis" that you can copy from any environment
- All sqlite db files from /var/lib/lpms and /var/lib/ccpark can be copied (if you don't want an environment from scratch) or they are generated automatically when you start lpms and ccpark services
- LPMS and CCPARK services that should run on the instance, they can be installed if you have the packages or compiled from our github: <https://github.com/QPcorp/ParkingSW>

- Configure the timezone on the server:

```
sudo dpkg-reconfigure tzdata
```

- Update all packages and clean whatever mess that might be on the server

```
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get autoremove && sudo apt-get autoclean && sudo apt-get purge && sudo apt-get clean && ntpdate ntp.ubuntu.com
```

- Install SSH server and basic useful packages

```
sudo apt-get install openssh-server sqlite3 gcc make nano git libconfuse0 unzip
```

- Add Deploy user

```
useradd -c "Deploy user" deploy -s /bin/bash
mkdir /home/deploy/
mkdir /home/deploy/.ssh
chmod 700 /home/deploy/.ssh
echo "-----BEGIN RSA PRIVATE KEY-----
MIIEEowIBAAKCAQEAyZiH4EGZSB0Zz9tA/CgItkSHk8iThzLQBtsPFXWzvNGjsAYD
oPPqXZ7+6BDQhZp18WVbEmR0AETREDorZnoYo3388cRwwuqLMjqyJprWBgtLtLp
jIwxRMM8GdSUK1ZQL4Yos6q0eKn8Kxn23yMbeQeRb+RfMk5JSIQmxCHGYrtiRuFX
l0dBimGd4tsQubd8d+Dqtg10tdPHZa5t/N+1BjG/A/zyENZN1BW2jGDNwsiv0wnf
Sb0oj5ecCE06KCGWHr-fvQGHeOu+v9JJlKtVXApqorDeB6g+83FHmW45W8h1cen1
UQzWwLzhD1hVUYGONP7JhxA12C/wjSR1S1HMwIDAQABAoIBAQC7sSiokUq+AYnO
Lut2uWtLD8jOvgSk8Bkgm0mVSE28uc5Vv303cgE9gx+/PpB2I4dQVdaR7DbYXTns
```

```
iswHkBVkdC2BhyyZPVojiDiX5rWUfHiqRLIvKj4Itt2EIh5QfStdDN/xDMubuBQ2D
r5vaqlwUtggghjVXX+2Q94EEUJHdLwQddSLRwWG6vEyUBQ1ESGJOJiqcNIwtIu2
uIFQL0BM5Kn13Yoauw08ElwiL0TNeCo5ksPQL8wMByeQryiAvE1U2Kxyx+MH44MZ
loUfTtS1+rQW/hJEN1GNMun9Hnqo0ZQPXYVA05rWqiktI5K3B8+pLgOJL+7RaYi
scG0cmBxAoGBAOoh9xK9RN+0d5c79q0f4WHuyz5qWhCYmiLuVepiAvLnn/Z5aGtB
sz17lNyTkaCdBa6fXwgH8WvCzX2FbGk6CqeYX2D5DzLncnSeQJGKuer4kUwVyUPf
zzTxj0sq27GaYKA8gRV0aYakFB08w1gawnesG/IDYz4E15npMQHvK+UZAoGBANxs
nsgK8xdT1WrtSH9NgQHKxb6cK1ki61pnup9rKGpecV+kVKUvDQLd2VvPmmLqUqUY
AZDDeCt10+vJpmEz81Yd1a7EXaIm+wxM6R3pa/9JCieyIk+e1vAm9go+WMRKsB5
QH67I5aqn5A+AZQSUrdRf/itOcfE0B1tAQsIZawrAoGAGg0FbLRagXmXBODLJaAy
d68WNI+nLcmM/O3/6UBWi6WNcuM3VxfMVb72k43n2jMeeruI21/m4NKgAqNh2KN4
pAU0N5n+suxhG9Cv5oHF8fv6Trjt1/uDFqT6CKcQIsbGkY2p3zVAuJZxFXefWyI/
GQmxY1A2LqRcKefkaA6wL6ECgYB1xXNw3qjTs9N7t10/MOHk2zqchUP7LDSa6SFF
XDPT4XIh7+Np/qGtDmu8KkgK4bgWjaUIe+gRB53kdXF4LhGtyXMKVdDRepJYGB/V
44nYdGYAn6TFddDRr392Kamz+s1KEZLsI4D4V2G55+df/gpEBexg6e0dV4L6tesA
whPQ0QKBgB5/xwN7wJZ046E+DPvFKqvNnCE40oB0EgG5SoFHWNVQxZcQdH+dvqnV
GnfhEx83g790ZrUxNYdVKm+qCm59zf0BzCmpnyQ60q+H6aiN2cKafywiEy1bpzta
28qWHP1Pf3htUXQ+ZyGs59UWCZZYkC0axfTVihdB62xwp/GDH5a2
-----END RSA PRIVATE KEY-----" > /home/deploy/.ssh/id_rsa
echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDJmIfgQZ1IE5nP20D8KAi2RIeTyJOHMTAG2w8Vdb080a0wBgOg8+pdnv7oENCFmmXxZVsSZHQARNEQOitm
chown -R deploy:deploy /home/deploy/
chmod 600 /home/deploy/.ssh/id_rsa
```

Add Users to Portal

```
useradd anai -m -s /bin/bash # We create anai user
mkdir -p /home/anai/ # We add the home folder in case it's not created
passwd anai # We change anai password
usermod -a -G www-data anai # We add anai to www-data group
usermod -a -G sudo anai # We add anai to sudo group

#while adding a new user
chmod 700 /home/chris/.ssh
chmod 600 /home/chris/.ssh/*
chown -R chris:chris /home/chris
```

Remove the need to type in password to become sudo

```
sudo visudo
Add this line for $user:
    $user ALL=(ALL) NOPASSWD:ALL
```

Installation of PHP

```
# Install required packages to compile php
sudo apt-get install apache2-threaded-dev apache2-mpm-prefork apache2-prefork-dev libcurl4-openssl-dev
sudo apt-get install libsqlite3-dev sqlite3 mysql-server libmysqlclient-dev libreadline-dev libzip-dev libxslt1-dev
sudo apt-get install libicu-dev libmcrypt-dev libmhash-dev libpcre3-dev libjpeg-dev libpng12-dev libfreetype6-dev libbz2-dev

# We get PHP Sources
wget http://museum.php.net/php5/php-5.3.10.tar.gz
tar xvf php-5.3.10.tar.gz
cd php-5.3.10

# We get a quick patch to avoid known bugs while compiling php 5.3.10
wget http://cpanelstuffs.linuxcabin.com/downloads/php1.patch
chmod 755 php1.patch
```

```

patch -p0 < php1.patch

# We configure, compile and install
./configure --help
./configure --with-apxs2=/usr/bin/apxs --with-mysql=mysqlnd --with-mysqli=mysqlnd --with-pdo-mysql=mysqlnd --with-openssl --wi
make clean
make
sudo make install

# Some recommendations
You may want to add: /usr/local/lib/php to your php.ini include_path
/home/anai/php-5.3.10/build/shtool install -c ext/phar/phar.phar /usr/local/bin
ln -s -f /usr/local/bin/phar.phar /usr/local/bin/phar
cp php.ini-production /usr/local/lib/php.ini
ln -s /usr/local/lib/php.ini /etc

# We tell apache how to find our freshly compiled php module
nano /etc/apache2/apache2.conf
# Use for PHP 5.x:
LoadModule php5_module modules/libphp5.so
AddHandler php5-script .php
# Add index.php to your DirectoryIndex line:
DirectoryIndex index.html index.php
AddType text/html .php
# PHP Syntax Coloring
# (optional but useful for reading PHP source for debugging):
AddType application/x-httpd-php-source phps

# Drupal likes to know where our mysqli socket is
mysql --print-defaults
Gives you: /var/run/mysqld/mysqld.sock
Update php.ini with mysqli.default_socket = /var/run/mysqld/mysqld.sock

```

Installation of Apache, Mysql and Drupal

```

# Install packages
apt-get install apache2 mysql-server mysql-client sqlite3 git libconfuse0
a2enmod rewrite

# Get Drupal 6.29 code
wget http://ftp.drupal.org/files/projects/drupal-6.29.tar.gz
tar zxvf drupal-6.29.tar.gz
mv drupal-6.29/* /home/anai/web/drupal/

# Enter the mysql command-line
mysql -u root -p
# Create a mysql db for drupal, don't forget to create a mysql user for it
CREATE DATABASE drupal CHARACTER SET utf8 COLLATE utf8_general_ci;
# Create a mysql user for it
CREATE USER 'webadmin'@'localhost' IDENTIFIED BY 'g7uHJAn9';
# Grant privileges
GRANT ALL PRIVILEGES ON * . * TO 'webadmin'@'localhost';
FLUSH PRIVILEGES;
quit

# Create missing folders and apply permissions
mkdir -p /home/anai/web/drupal/sites/all/modules
mkdir -p /home/anai/web/drupal/sites/all/themes
mkdir -p /home/anai/web/drupal/sites/default/files
chmod 770 /home/anai/web/drupal/sites/default/files

```

```
# Copy default configuration, settings will be asked when you access the drupal for the first time
cp /home/anai/web/drupal/sites/default/default.settings.php /home/anai/web/drupal/sites/default/settings.php
sudo chmod a+w /home/anai/web/drupal/sites/default/settings.php
cd /home/anai/web/drupal/sites/all/modules/
mkdir d6
chown -R deploy:deploy /home/anai/web/drupal/sites/all/modules/d6/
cd d6/
su deploy

# Deploy d6 modules
git clone git@github.com:QPcorp/d6.git .

exit
```

Add to /etc/apache2/envvars:

```
## LPMS variables
export APPS_LPMS_AGENT_DB=/var/lib/lpms/agent.db
export APPS_LPMS_LPNMASTER_DB=/var/lib/lpms/master.db
export APPS_LPMS_MASTER_DB=/var/lib/lpms/master.db
export APPS_LPMS_LPN_DB=/var/lib/lpms/lpn.db
export APPS_LPMS_LPIMG_DB=/var/lib/lpms/lpimg.db
export APPS_LPMS_CARIMG_DB=/var/lib/lpms/carimg.db
export APPS_LPMS_PROP_DB=/var/lib/lpms/prop.db
export APPS_LPMS_REPOSSESS_DB=/var/lib/lpms/reposseess.db
export APPS_LPMS_ADMIN_DB=/var/lib/lpms/admin.db
export APPS_LPMS_PERMIT_DB=/var/lib/lpms/permit.db
export APPS_LPMS_ALERT_DB=/var/lib/lpms/alert.db
export APPS_LPMS_COMPLAINT_DB=/var/lib/lpms/complaint.db
export APPS_LPMS_TICKET_DB=/var/lib/lpms/ticket.db
export APPS_LPMS_BOOT_DB=/var/lib/lpms/boot.db
export APPS_LPMS_TOWING_DB=/var/lib/lpms/towing.db
export APPS_LPMS_NOTIFY_DB=/var/lib/lpms/notify.db
export APPS_LPMS_STATS_DB=/var/lib/lpms/stats.db
export APPS_LPMS_FREQ_USERS_DB=/var/lib/lpms/freq_users.db
export APPS_LPMS_VALET_DB=/var/lib/lpms/valet.db
export APPS_LPMS_HOTLIST_DB=/var/lib/lpms/hotlist.db
export APPS_LPMS_OCCU_DB=/var/lib/lpms/occu.db
export APPS_LPMS_HISTORY_DB=/var/lib/lpms/history.db
## CCPARK variables
export APPS_CCPARK_CONTROL_DB=/var/lib/ccpark/control.db
export APPS_CCPARK_ALARMS_DB=/var/lib/ccpark/alarms.db
export APPS_CCPARK_AUDITS_DB=/var/lib/ccpark/audits.db
export APPS_CCPARK_CASH_DB=/var/lib/ccpark/cash.db
export APPS_CCPARK_SPACE_DB=/var/lib/ccpark/space.db
export APPS_CCPARK_STATS_DB=/var/lib/ccpark/stats.db
export APPS_CCPARK_CLIENT_DB=/var/lib/ccpark/clientdata.db
export APPS_CCPARK_HISTORY_DB=/var/lib/ccpark/history.db
export APPS_CCPARK_EMPLOYEE_DB=/var/lib/ccpark/employee.db
export APPS_CCPARK_TICKET_DB=/var/lib/ccpark/ticket_dispenser.db
```

Add a vhost to /etc/apache2/sites-available/<portal_name>.conf:

```
<VirtualHost *:80>
ServerAdmin paulc@locomobi.com
ServerName <portal_name>.locomobi.com
DocumentRoot /home/anai/web/drupal
<Directory />
    Options FollowSymLinks
    AllowOverride All
</Directory>
ErrorLog ${APACHE_LOG_DIR}/<portal_name>-error.log
CustomLog /var/log/apache2/<portal_name>-access.log combined
```



```
# Transfer/SCP the files to your target instance

# Unzip the files
tar xvzf themes.tar.gz -C /var/www/drupal/sites/all/themes/
tar xvzf lpms.tar.gz -C /var/lib/lpms/
tar xvzf ccpark.tar.gz -C /var/lib/ccpark/
```

Install lpms and ccpark services

Build the software (usually should be done off-portal then scp'd over)

```
First get the source: git clone git@github.com:QPcorp/ParkingSW
Next set up the environment variables: bash bash_setup.sh
Now build the apps directories, one by one:
cd apps/argparse/mak
make clean;make
cd ../../bml/mak
make clean;make
cd ../../common
make clean;make
cd ../../cpp
make clean;make
cd ../../fifo
make clean;make
cd ../../ipc
make clean;make
cd ../../pmem
make clean;make
cd ../../socket
make clean;make
cd ../../sqldb
make clean;make
cd ../../verbose
make clean;make
Now, build ccpark:
cd ../../../../ccpark/mak
make clean;make bin;strip ../bin/ccpark
Rename the ccpark to ccpark.bin with the versionname as found in the ../inc/ccpark.h file:
cp ../bin/ccpark ../bin/ccpark.bin.VERSIONNAME
Now, build lpms:
cd ../../lpms/mak
make clean;make bin;strip ../bin/lpms
Rename the lpms to lpms.bin with the versionname as found in the ../inc/lpms.h file:
cp ../bin/lpms ../bin/lpms.bin.VERSIONNAME
```

Copy relevant files over to the portal and place them in the correct locations

```
From the ParkingSW directory, do:
scp ccpark/bin/ccpark anai@portalname:~/
scp lpms/bin/lpms anai@portalname:~/
tar zcvf ccpark_conf.tar.gz ccpark/cfg/*
scp ccpark_conf.tar.gz anai@portalname:~/
tar zcvf lpms_conf.tar.gz lpms/cfg/*
scp lpms_conf.tar.gz anai@portalname:~/
Login to the portal as anai. Then extract the tar files.
tar zxvf ccpark_conf.tar.gz
tar zxvf lpms_conf.tar.gz
First make the proper directories:
sudo mkdir /usr/lib/ccpark
sudo mkdir /usr/lib/lpms
sudo mkdir /etc/ccpark
sudo mkdir /etc/lpms
```

Then copy the files over:

```
sudo cp ccpark.bin.VERSIONNAME /usr/lib/ccpark
sudo cp lpms.bin.VERSIONNAME /usr/lib/lpms
sudo cp ccpark/cfg/* /etc/ccpark
sudo cp lpms/cfg/* /etc/lpms
```

Finally, make the symlinks:

```
sudo ln -s /usr/lib/ccpark/ccpark.bin.VERSIONNAME /usr/lib/ccpark/ccpark
sudo ln -s /usr/lib/lpms/lpms.bin.VERSIONNAME /usr/lib/lpms/lpms
```

Get the packages and install them (deprecated instructions)

deb files are available here: <https://github.com/QPcorp/nautical-vagrant-box/tree/master/automation/puppet/files/anai-services>
 dpkg -i anai-ccpark.deb
 dpkg -i anai-lpms.deb

Start lpms and ccpark services (deprecated instructions)

```
service lpms start && service ccpark start
```

Permissions

```
mkdir -p /home/anai/web/drupal/tmpimage/
chmod 770 -R /home/anai/web/drupal/tmpimage/
chown -R www-data:www-data /home/anai/web/drupal/
chmod -R 775 /var/lib/lpms
chmod -R 775 /var/lib/ccpark
chown -R root:www-data /var/lib/lpms
chown -R root:www-data /var/lib/ccpark
```

Configure the ports

```
sudo vi /etc/ccpark/socket.conf
Change the configuration starting with listen_port to the assigned port for ccpark.
sudo vi /etc/lpms/socket.conf
Change the configuration starting with listen_port to the assigned port for lpms.
```

Set up the upstart

The following two files need to be copied from the ParkingSW repository to the /etc/init directory:

/etc/init/ccpark.conf

```
# upstart script for ccpark
# location: /etc/init/ccpark.conf
# usage:
# sudo start ccpark
# sudo stop ccpark
start on stopped rc RUNLEVEL=[2345] and (
    not-container or
    container CONTAINER=lxc or
    container CONTAINER=lxc-libvirt)

stop on runlevel [!2345]
exec /usr/lib/ccpark/ccpark
respawn

post-start script
echo "`date +%F %T`: CCPARK service started. " >> /var/log/ccpark-upstart.log
```

```
echo "CCPARK Started at `date +%F %T` on `hostname`" | mail -s "CCPARK Started on `hostname`" 'sanket@qpme.com, daniel@1
end script
```

/etc/init/lpms.conf

```
# upstart script for lpms
# location: /etc/init/lpms.conf
# usage:
# sudo start lpms
# sudo stop lpms
start on stopped rc RUNLEVEL=[2345] and (
    not-container or
    container CONTAINER=lxc or
    container CONTAINER=lxc-libvirt)

stop on runlevel [!2345]
exec /usr/lib/lpms/lpms
respawn

post-start script
    echo "`date +%F %T`: LPMS service started. " >> /var/log/lpms-upstart.log
    echo "LPMS Started at `date +%F %T` on `hostname`" | mail -s "LPMS Started on `hostname`" 'sanket@qpme.com, daniel@locom
end script
```

Run lpms and ccpark to create .db files

```
start lpms
start ccpark
Verify they are running properly:
tail -f -n 250 /var/lib/ccpark/ccpark.log
tail -f -n 250 /var/lib/lpms/lpms.log
```

Set up scripts

Scripts go into the /home/anai/scripts folder:

drupal_cleaner.sh

```
#!/bin/bash
# DRUPAL CLEANER
# This script helps to clean a server with drupal and anai services ccpark and lpms
MYSQL_ROOT_USER="root"
MYSQL_ROOT_PASSWORD="thepassword"
MYSQL_DATABASE="lpms"

# Truncate lpn tables older than a month
echo "Truncate lpn tables older than a month"
find /var/lib/lpms/ -name "lpn_*" -mtime +31 | while read fname; do
    /usr/lib/sqlite3 $fname < /var/lib/lpms/truncate.sql
done

# Remove tmp images older than a month
echo "Delete tmp images older than a month..."
find /home/anai/web/drupal/tmpimage/ -mtime +31 -exec rm '{}' \;

# Clean ccpark and LPMS older logs
find /var/lib/lpms/ -iname "lpms.log.0[1-9]*" -exec rm '{}' \;
```



```
find /var/lib/ccpark/ -iname "ccpark.log.0[1-9]*" -exec rm '{}' \;
```

```
# Clear drupal portal cache_form table
```

```
echo "Truncate drupal portal cache_form table..."
```

```
mysql -i u$MYSQL_ROOT_PASSWORD $MYSQL_DATABASE -e'DELETE FROM cache_form';
```

```
echo "Done!"
```

lpn_db_daily_check.sh

```
TODAY=$(date + "%Y-%m-%d")
FILENAME=/var/lib/lpms/lpn_$TODAY.db
SQL_FILENAME=/var/lib/lpms/create_table_lpms_daily_lpn_db.sql
```

```
if [ -f $FILENAME ];
then
    FILESIZE=$(stat -c%s "$FILENAME")
    if [ $FILESIZE -eq 0 ];
    then
        echo "`date + "%F %T"` - File $FILENAME exists, it's size is 0."
        echo "`date + "%F %T"` - Executing SQL file on $FILENAME to create tables..."
        sqlite3 $FILENAME < $SQL_FILENAME
        echo "`date + "%F %T"` - Done."
        echo "`date + "%F %T"` - Restarting LPMS..."
        /sbin/restart lpms
        echo "`date + "%F %T"` - Done."
    else
        echo "`date + "%F %T"` - File $FILENAME exists, it's size is $FILESIZE"
    fi
else
    echo "`date + "%F %T"` - File $FILENAME does not exist."
fi
```

daily_backup.sh

```
#!/bin/bash
```

```
# Make sure you have the S3 bucket locomobi-backups created, and a folder inside with the name of your portal.
```

```
# Settings
```

```
DATE=$(date + "%d-%b-%Y")
DRUPAL_FILE_PATH="/home/anai/web/drupal"
DRUPAL_DB_HOST="localhost"
DRUPAL_DB_NAME="lpms"
DRUPAL_DB_USER="webadmin"
DRUPAL_DB_PASSWORD="g7uHJAn9"
TEMPORARY_BACKUP_PATH="/tmp"
PORTAL_NAME="portal-name-goes-here"
```

```
# Backup lpms and ccpark database files
```

```
tar cvzf $TEMPORARY_BACKUP_PATH/lpms-$DATE.tar.gz /var/lib/lpms/*.db
```

```
tar cvzf $TEMPORARY_BACKUP_PATH/ccpark-$DATE.tar.gz /var/lib/ccpark/*.db
```

```
rm /var/lib/ccpark/ccpark*aws
```

```
for f in /var/lib/ccpark/ccpark.log*
do
    grep "+++" $f > "$f.aws"
done
```

```
tar cvzf $TEMPORARY_BACKUP_PATH/ccpark_logs-$DATE.aws.tar.gz /var/lib/ccpark/*.aws
```

```
# Backup drupal database files
mysqldump --user=$DRUPAL_DB_USER --password=$DRUPAL_DB_PASSWORD --host=$DRUPAL_DB_HOST --databases $DRUPAL_DB_NAME > $TEMPORAR
tar cvzf $TEMPORARY_BACKUP_PATH/drupal-db-backup-$DATE.tar.gz $TEMPORARY_BACKUP_PATH/drupal-db-backup-$DATE.sql

# Backup customer portal files
#tar cvzf $TEMPORARY_BACKUP_PATH/portal-files-backup-$DATE.tar.gz $DRUPAL_FILE_PATH/

# Transfer all the files to the folder assigned for that portal
/usr/bin/s3cmd -f -c /home/anai/scripts/.s3cfg sync $TEMPORARY_BACKUP_PATH/*.gz s3://locomobi-backups/$PORTAL_NAME/

/usr/bin/s3cmd -f -c /home/anai/scripts/.s3cfg sync $TEMPORARY_BACKUP_PATH/*.aws*.gz s3://defero-logs/$PORTAL_NAME/ccpark/

# Clean backup files
rm -f $TEMPORARY_BACKUP_PATH/*.sql
rm -f $TEMPORARY_BACKUP_PATH/*.gz
```

Permissions for scripts

```
sudo chmod 644 drupal_cleaner.sh
sudo chmod 644 daily_backup.sh
```

Set up S3 configuration file .s3cfg:

```
.s3cfg

[default]
access_key = AKIAJ4RL05BZDC7VLDLQ
secret_key = XWPE/xmP6SRoJ7IEaBRIJVXR69muEvFKpOF4MB0V
use_https = True
```

Set up domains and Reverse Proxy

- Login to AWS console and go to Route 53 by clicking [here](#).
- Create Two new A records as following
 - Name: PARTNER.locomobi.com, Type: A-IPV4 address, Value: 104.153.58.174
 - Name: api-PARTNER.locomobi.com, Type: A-IPV4 address, Value: 104.153.58.174
- ssh in to Reverse Proxy, credentials are [here](#).
- perform following steps
 - cd /etc/apache2/sites-available
 - copy lefrak.locomobi.com PARTNER.locomobi.com
 - edit PARTNER.locomobi.com and change following variables
 - ServerName = PARTNER.locomobi.com
 - ProxyPass and ProxyPassReverse to the Internal IP of new server
 - copy api-lefrak.locomobi.com api-PARTNER.locomobi.com
 - edit api-PARTNER.locomobi.com and change following variables
 - ServerName = api-PARTNER.locomobi.com
 - ProxyPass and ProxyPassReverse to the Internal IP of new server
- run following commands
 - a2ensite PARTNER.locomobi.com
 - a2ensite api-PARTNER.locomobi.com
 - service apache2 reload

Finish your installation

- Go to your Drupal website at <portal_name>.locomobi.com
- Follow prompts to complete Drupal for any errors/missing data.
- Check all ANAI modules are enabled.
- Assign Roles Admin, Enforcer, Manager, Security, Tenant ? These are drupal user roles assigned at time of user setup
- Assign Admin -> Modules -> FrontPage to be the same as the environment you copy ? Frontpage should be chpfront for authenticated users
- Set correct listening ports in /etc/lpms/socket.conf and /etc/ccpark/socket.conf ? These are the services LPMS controls cameras, CCPARK controls pay kiosks. Because of Cloud Firewall etc these services need to be configured to listen on their own distinct ports
- MySQL server is purely for Drupal. Services use SQLite3 databases.

- Go to Drupal at <portal_name>.locomobi.com/admin/settings/site-information and update the site name.
- Via ssh, go to /home/anai/web/drupal-6.29/sites/default/files/ and copy over the logo file to metropolis_logo.png.
- If admin page is slow, add FQDN to /etc/hosts

```
# Quick Fix for MerchantCode
sqlite3 /var/lib/lpms/stats.db ".schema reportstats_table"
sqlite3 /var/lib/lpms/stats.db "ALTER TABLE reportstats_table ADD COLUMN MerchantCode DEFAULT 'ANY';"
sqlite3 /var/lib/lpms/stats.db ".schema reportstats_table"
```

Automate import of sqlite database from production environments (ideal for testing environments)

```
apt-get install s3cmd
mkdir -p /home/anai/scripts/
```

nano /home/anai/scripts/import_db.sh

```
#!/bin/bash
LOGFILE=import_db.log
exec 3>&1 1>>${LOGFILE} 2>&1
NOW=$(date +"%k:%M %m-%d-%Y")
echo "
-----
Import Sqlite Backups from Production environments
Requirements:
- s3cmd package
- .s3cfg configuration file in the same folder
Export to /var/lib/lpms & ccpark (Replace method)
-----
[1] Hotel2
[2] Psmart2
[3] Stpats
[4] Trca
[5] Locomobi
[6] Samuels
[7] TTSI
[8] Simplytechs
[9] Joes
[10] Resort2
[11] Eastongroup
[12] WallyPark
[13] BigCypress
Please enter your choice:
" 1>&3
read option
case $option in
1)
    BACKUP_PATH="locomobi-backups/hotel2"
    ;;
2)
    BACKUP_PATH="locomobi-backups/psmart2"
    ;;
3)
    BACKUP_PATH="locomobi-backups/stpats"
    ;;
4)
    BACKUP_PATH="locomobi-backups/trca"
    ;;
5)
    BACKUP_PATH="locomobi-backups/locomobi"
```

```

;;
6)
  BACKUP_PATH="locomobi-backups/samuels"
  ;;
7)
  BACKUP_PATH="locomobi-backups/ttsi"
  ;;
8)
  BACKUP_PATH="locomobi-backups/simplytechs"
  ;;
9)
  BACKUP_PATH="locomobi-backups/joes"
  ;;
10)
  BACKUP_PATH="locomobi-backups/resorts2"
  ;;
11)
  BACKUP_PATH="locomobi-backups/eastongroup"
  ;;
12)
  BACKUP_PATH="locomobi-backups/wallypark"
  ;;
13)
  BACKUP_PATH="locomobi-backups/bigcypress"
  ;;
esac
echo "-----" | tee /dev/fd/3
echo "$NOW: Importing CCPARK backup from $BACKUP_PATH..." | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
s3cmd -f -c .s3cfg get `s3cmd -c .s3cfg ls s3://$BACKUP_PATH/ccpark* | sort -n | tail -1 | awk '{print $4}` ccpark-latest-back
echo "Done!" | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
echo "$NOW: Importing LPMS backup from $BACKUP_PATH..." | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
s3cmd -f -c .s3cfg get `s3cmd -c .s3cfg ls s3://$BACKUP_PATH/lpms* | sort -n | tail -1 | awk '{print $4}` lpms-latest-backup.
echo "Done!" | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
echo "$NOW: Replacing CCPARK with backup from $BACKUP_PATH..." | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
tar xvzf ccpark-latest-backup.tar.gz -C / | tee /dev/fd/3
echo "Done!" | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
echo "$NOW: Replacing LPMS with backup from $BACKUP_PATH..." | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
tar xvzf lpms-latest-backup.tar.gz -C / | tee /dev/fd/3
echo "Done!" | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
echo "$NOW: Cleaning files..." | tee /dev/fd/3
echo "-----" | tee /dev/fd/3
rm ccpark-latest-backup.tar.gz && rm lpms-latest-backup.tar.gz | tee /dev/fd/3
echo "Done!" | tee /dev/fd/3

```

nano /home/anai/scripts/.s3cfg

```

[default]
access_key = <AWS S3 Access key>
secret_key = <AWS S3 Secret Key>
use_https = True

```

To use the script just type: `sh /home/anai/scripts/import_db.sh` and select the portal you want to import from.

Clean automatically your LPMS/CCPARK/Drupal environment

nano /home/anai/scripts/drupal_cleaner.sh

```
#!/bin/bash
# DRUPAL CLEANER
# This script helps to clean a server with drupal and anai services ccpark and lpms
MYSQL_ROOT_USER="<To complete>"
MYSQL_ROOT_PASSWORD="<To complete>"
MYSQL_DATABASE="<To complete>"
# Truncate lpn tables older than a month
echo "Truncate lpn tables older than a month..."
find /var/lib/lpms/ -name "lpn_*" -mtime +31|while read fname; do
  /usr/bin/sqlite3 $fname < /var/lib/lpms/truncate.sql
done
# Remove tmp images older than a month
echo "Delete tmp images older than a month..."
find /var/www/drupal/tmpimage/ -mtime +31 -exec rm '{}' \;
# Clean CCpark and LPMS older logs
echo "Delete CCpark and LPMS old logs..."
find /var/lib/lpms/ -iname "lpms.log.0[1-9]*" -exec rm '{}' \;
find /var/lib/ccpark/ -iname "ccpark.log.0[1-9]*" -exec rm '{}' \;
# Clear drupal portal cache_form table
echo "Truncate drupal portal cache_form table..."
mysql -u$MYSQL_ROOT_USER -p$MYSQL_ROOT_PASSWORD $MYSQL_DATABASE -e'DELETE FROM cache_form';
echo "Done!"
```

Add the following cronjob to your system

```
0 0 * * * /home/anai/scripts/drupal_cleaner.sh >> /var/log/drupal_cleaner.log 2>&1
```

git@github.com:QPcorp/d6.git

2 Comments



Jefferson To

tar cvzf theme.tar.gz --directory=/home/anai/web/drupal/themes/metropolis/ .



Jefferson To

zip and copy jquery.ui folder from /home/anai/web/drupal/sites/libraries