**Цель:**  Создать процесс выполняющий периодическое обновление адресной базы СРМ "Смородина Диалог", использующий данные скачанные с адресного реестра ФНС.

**Термины:**

1) Целевая папка - в ней размещается архивы и после выполнения ежесуточных обновлений,

появляются протоколы обновления.

2) Владелец - пользователь для которого создаётся расписание CRONTAB. Условно,

это <user> группа <users>.

3) Пользователь БД - должен обладать правами на выполнение в БД операций типа CRUD.

**Что необходимо:**

1) BASH-скрипт "load\_updates.sh".

2) Расписание "crontab.txt".

**Основные этапы установки:**

1) Отредактировать "load\_updates.sh".

2) Создать папку "/var/log/db\_upgrade", установить владельца для "db\_upgrade".

3) Взять файл с очередными обновлениями и положить его в целевую папку.

4) Запустить его из командной строки, убедиться в том, что обновление прошло

успешно.

5) Создать расписание для CRONTAB, положить очередное обновление в целевую папку.

На следующий день просмотреть результаты обновления и убедиться в том, что

обновление прошло успешно.

**Как происходит обновление:**  Архив с обновления адресной базы помещается в целевую папку. Например:

целевая папка "/tmp"?, архив "up\_prds\_20230206.tar.gz",

имя архива строится по конкретным правилам:

"up\_prds\_" - префикс

".tar.gz" - суффикс

"2023" - год

"02" - месяц

"06" - дата

Если изменить префикс, либо суффикс, то процесс обновления не запустится.

Процесс обновления адресной базы должен запускаться один раз, в вечернее время.

На следующий день состояние файлов в целевой папке изменится:

- архив исчезнет

- вместо него появятся два файла, содержащих протокол загрузки:

"up\_prds\_20230306.out"

up\_prds\_20230306.err"

Ниже приведён фрагмент файла "up\_prds\_20230306.out"

DATA/adr\_area\_87.sql

BEGIN

DELETE 3

COPY 5

COMMIT

DATA/adr\_street\_02.sql

BEGIN

DELETE 0

COPY 1

COMMIT

DATA/adr\_street\_03.sql

BEGIN

DELETE 0

COPY 0

COMMIT

DATA/adr\_house\_04.sql

BEGIN

DELETE 9

COPY 9

COMMIT

Сначала идут идут обновления адресных пространств, потом улиц, потом домов.

Свидетельством того, что процесс обновления идёт правильно, является наличие пар

BEGIN -- COMMIT Если пояляется пара BEGIN -- ROLLBACK, то это означает, что для

данного адресного региона (цифра в имени файла) обновление не было выполнено.

Причину ошибки можно посмотреть в файле типа "err".

Ниже приведён фрагмент файла "up\_prds\_20230306.err"

Sat Mar 11 22:00:03 MSK 2023

DATA/adr\_area\_02.sql

DATA/adr\_area\_03.sql

DATA/adr\_area\_04.sql

DATA/adr\_area\_05.sql

В обычном состоянии он содержит перечень обработанных файлов с региональными обновлениями адресной базы. ВНИМАНИЕ номер адресного региона в имени файла соответствует стандартам адресной базы, и не совпадают с нумерацией регионов в адресном реестре ФНС.

В том случаем, если в процессе обновления возникла ошибка, она зафиксируется в

соответствующем месте файла типа "err".

**Установка:**

**1) Настройка файла "load\_updates.sh".**  Она выполняется один раз, перед размещением файла

в каталоге "/usr/local/bin" и в дальнейшем используется для выполнения обновления

обновления адресной базы.

В файле два блока, они отмечены комментариями, содержимое блоков необходимо отредактировать.

Первый блок.

## ----------------------------------------------------- ##

## Общие переменные устанавливать только вручную ##

## ----------------------------------------------------- ##

**WD=/tmp** # Рабочий каталог, сюда должен попасть архив с обновлениямим

Z5=$(ls /tmp/up\_prds\_\*tar.gz.md5)

ZM=$(ls -tr1 /tmp/up\_prds\_\*tar.gz)

if [ $? -ne 0 ]

then

exit 1

fi

# Маска имени архива ???

**IP="<\_EXAMPLE\_127.0.0.1>" # IP сервера БД**

**PT="<\_EXAMPLE\_5432>" # Порт сервера БД**

**UM="<\_EXAMPLE\_postgres>" # Имя пользователя БД**

**DB="<\_EXAMPLE\_unnsi\_prd>" # Имя базы**

## ----------------------------------------------------- ##

## УСТАНОВИТЬ И НЕ МЕНЯТЬ БЕЗ ОСОБОЙ НЕОБХОДИМОСТИМ ##

## ----------------------------------------------------- ##

Обязательно необходимо установить значения для перемеменных IP, PT, UM, DB

Для имени пользователя БД, необходимо создать запись в файле ".pgpass

Второй блок. Переменная ZD должна менять значения строго в соответствии с значениями переменных Z5, ZM. МЕНЯТЬ ЕЁ ЗНАЧЕНИЕ ТОЛЬКО В СЛУЧАЕ КРАЙНЕ НЕОБХОДИМОСТИ.

## ----------------------------------------------------- ##

## Общие переменные устанавливать только вручную ##

## ----------------------------------------------------- ##

ZD=$(ls -dtr1 /tmp/up\_prds\_????????) # Только каталоги.

if [ $? -ne 0 ]

then

exit 1

fi

## ----------------------------------------------------- ##

## УСТАНОВИТЬ И НЕ МЕНЯТЬ БЕЗ ОСОБОЙ НЕОБХОДИМОСТИМ ##

## ----------------------------------------------------- ##

Далее файл размещается в папке "/usr/local/bin".

Для Владельца и группы даются права на выполнение.

$> chmod ug=+x+r+w /usr/local/bin/load\_updates.sh

**2) Создать папку для размещения log-файла.**

$> sudo mkdir /var/log/db\_upgrade

$>sudo chown <user>.<users> /var/log/db\_upgrade

Выполняем тестовый запуск из командной строки.

/usr/local/bin/load\_updates.sh 1>> /var/log/db\_upgrade/upgrade.out 2>> /var/log/db\_upgrade/upgrade.err

В том случае, если в целевой папке нет архива обновлениями, в log-файле "upgarde.err" появится

строка

ls: cannot access /tmp/up\_prds\_\*tar.gz: No such file or directory

**3) Cоздать расписание CRONTAB.**

Расписание создаётся для пользователя-Владельца имеющего права на выполнение /usr/local/bin/load\_updates.sh

Файл "crontab.txt".

# Edit this file to introduce tasks to be run by cron.

#

# Each task to run has to be defined through a single line

# indicating with different fields when the task will be run

# and what command to run for the task

#

# To define the time you can provide concrete values for

# minute (m), hour (h), day of month (dom), month (mon),

# and day of week (dow) or use '\*' in these fields (for 'any').#

# Notice that tasks will be started based on the cron's system

# daemon's notion of time and timezones.

#

# Output of the crontab jobs (including errors) is sent through

# email to the user the crontab file belongs to (unless redirected).

#

# For example, you can run a backup of all your user accounts

# at 5 a.m every week with:

# 0 5 \* \* 1 tar -zcf /var/backups/home.tgz /home/

#

# For more information see the manual pages of crontab(5) and cron(8)

#

# m h dom mon dow command

0 22 \* \* \* /usr/local/bin/load\_updates.sh 1>> /var/log/db\_upgrade/upgrade.out 2>> /var/log/db\_upgrade/upgrade.err

Т.е каждый день в 22-00 будет выполняется команда.

/usr/local/bin/load\_updates.sh 1>> /var/log/db\_upgrade/upgrade.out 2>> /var/log/db\_upgrade/upgrade.err

**Создание расписания:**

$> crontab -l -- контрольный запрос

$> crontab crontab.txt -- создание расписания.