

Table of Contents

2 Администрирование БД	3
Активация встроенной БД	3
Увеличение размера БД	3
Восстановление БД с переносом данных	4
Удаление БД и загрузка в нее данных заново из внешнего источника (билинга и т.п.)	4
Экспериментальный раздел	5
Восстановление БД в формат команд fdpi_ctrl	5
Восстановление отдельных таблиц БД в формат команд fdpi_ctrl	5

2 Администрирование БД

UDR (встроенная база данных, user data repository) предназначена для постоянного хранения данных об услугах и настройках полисинга для абонентов оператора.

Список таблиц в БД

Таблица	Назначение
bindings	Связь login и адреса
bindings_multi	Связь login и адреса для мультиабонентов (с несколькими IP)
policing	Настройки полисинга для абонентов
profile_names	Имена профилей
profiles	Профили услуг и полисинга
services	Настройки услуг для абонентов
vchannel_policing	Настройки полисинга для каналов
ip_props	Свойства абонентов BRAS

Активация встроенной БД



Активация UDR требуется в обязательном порядке при использовании динамических IP адресов и для поддержки абонентов с несколькими IP.

Активация UDR осуществляется настроенным параметром в файле **/etc/dpi/fastdpi.conf**

```
udr=1
```

Созданная база данных размещается в каталоге **/var/db/dpi**



Сделать копию БД можно без остановки фрі командой резервного копирования

```
mdb_copy /var/db/dpi ./DB
```

и выполнять дальнейшие манипуляции над копией БД.

Увеличение размера БД

По умолчанию размер БД ограничен 1GB, если у вас количество профилей полисинга превышает 1 млн, то потребуется увеличить размер по умолчанию:

```
udr_size=2147483648
```

установит размер БД в 2GB



Встроенная БД не требует администрирования и устойчива к отказам. Тем не менее в редких случаях происходило повреждение встроенной БД. В этом случае возможны следующие варианты действий:

Восстановление БД с переносом данных

Останавливаем dpi

```
service fastdpi stop
```

Выполняем скрипт

```
rm -rf /var/db/dpi.recover/*
mkdir -p /var/db/dpi.recover/tmp
for table in $(mdb_dump -l /var/db/dpi); do
  mdb_dump -f /var/db/dpi.recover/tmp/dump.$table.load -s $table /var/db/dpi
  mdb_load -f /var/db/dpi.recover/tmp/dump.$table.load /var/db/dpi.recover
done
rm /var/db/dpi/lock.mdb
mv /var/db/dpi/data.mdb /var/db/dpi.recover/data.mdb.backup
cp -f /var/db/dpi.recover/data.mdb /var/db/dpi/
```

Запускаем dpi

```
service fastdpi start
```

Удаление БД и загрузка в нее данных заново из внешнего источника (билинга и т.п.)

Останавливаем dpi

```
service fastdpi stop
```

Удаляем БД

```
/bin/rm /var/db/dpi/*
```

Запускаем dpi

```
service fastdpi start
```

Загружаем в БД заново все настройки с помощью своих скриптов

Экспериментальный раздел

Восстановление БД в формат команд fdpi_ctrl

Останавливаем dpi

```
service fastdpi stop
```

Выполняем скрипт

```
mdb_dump -p -a -f dump.sh /var/db/dpi  
/bin/rm /var/db/dpi/*
```

Запускаем dpi

```
service fastdpi start
```

Запускаем скрипт

```
chmod +x dump.sh  
./dump.sh
```

Восстановление отдельных таблиц БД в формат команд fdpi_ctrl

Останавливаем dpi

```
service fastdpi stop
```

Выполняем скрипт

```
for table in $(mdb_dump -l /var/db/dpi); do  
mdb_dump -p -f dump.$table.sh -s $table /var/db/dpi  
done  
/bin/rm /var/db/dpi/*
```

Запускаем dpi

```
service fastdpi start
```

Выбираем нужные нам скрипты и запускаем, **пример**

```
chmod +x dump.bindings.sh  
./dump.bindings.sh
```