

# Table of Contents

<b>Database administration</b> .....	3
Activating the built-in database .....	3
Enlarging the DB .....	3
Database recovery with data transfer .....	4
Deleting a database and reloading data back from an external source (billing, etc.) .....	4
<b>Experimental Section</b> .....	5
Restoring a database to the fdpi_ctrl command format .....	5
Restoring certain tables to the fdpi_ctrl command format .....	5



# Database administration

UDR (built-in database, user data repository) is used for permanent storage of data on services and policing settings for the subscribers.

## List of database tables

Table	Purpose
bindings	login and address binding
bindings_multi	login and address binding for multi-subscribers (with several IPs)
policing	Subscriber policing setting
profile_names	Names of profiles
profiles	Profiles of services and policing
services	Subscriber services setting
vchannel_policing	Channel policing setting
ip_props	BRAS subscriber properties

## Activating the built-in database



UDR activation is required to support dynamic IPs and subscribers with several IPs.

UDR is activated by the configuration parameter in */etc/dpi/fastdpi.conf* file.

```
udr=1
```

The created database is located in the */var/db/dpi* directory



You can make a copy of the database without stopping the DPI with a backup command

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

and handle further manipulations in the copied database.

## Enlarging the DB

By default, the database size is limited to 1GB. If you have more than 1 million policing profiles, you will need to increase the default size:

```
udr_size=2147483648
```

will set the DB size of 2GB.



The built-in database does not require administration and is fault tolerant. However, in rare cases, damage to the embedded database occurred. In this case, one of the following options is possible.

## Database recovery with data transfer

Stop the fastDPI

```
service fastdpi stop
```

Run the script

```
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/
```

Start the fastDPI

```
service fastdpi start
```

## Deleting a database and reloading data back from an external source (billing, etc.)

Stop the fastDPI

```
service fastdpi stop
```

Delete the DB

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

Start the fastDPI

```
service fastdpi start
```

Reload all the settings into the database using own scripts.

# Experimental Section

## Restoring a database to the fdpi\_ctrl command format

Stop the fastDPI

```
service fastdpi stop
```

Run the script

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

Start the fastDPI

```
service fastdpi start
```

Run the script

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

## Restoring certain tables to the fdpi\_ctrl command format

Stop the fastDPI

```
service fastdpi stop
```

Run the script

```
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/*
```

Start the fastDPI

```
service fastdpi start
```

Choose and run the scripts you need, **for example**

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