

Системные требования.

- На сервере (здесь и далее примем, что “сервером” может быть как отдельный компьютер, так и виртуальная машина, VPS/VDS и т.д. должна быть установлена операционная система Linux семейства Debian не ниже 7.0 (Ubuntu 14.04 и выше), CentOS не ниже 6.5 или другие дистрибутивы с актуальной базой репозитория.
- IP ATC Asterisk должна быть версии не ниже 1.8.
На Asterisk должна быть настроена запись информации о разговорах в таблицы cel, cdr и queue_log, подробную информацию о необходимых настройках Asterisk вы найдете ниже в разделе «Настройки Asterisk».
- Сервер БД MySQL не ниже версии 5.5.

Настройки Asterisk

1. Создадим БД и таблицы в MySQL для нашей Asterisk

Для этого нужно войти в консоль MySQL и выполнить следующие запросы

```
CREATE DATABASE asteriskcdrdb;
USE asteriskcdrdb;

CREATE TABLE IF NOT EXISTS `cdr` (
  `id` INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,
  `calldate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `clid` VARCHAR(80) NOT NULL DEFAULT '',
  `src` VARCHAR(80) NOT NULL DEFAULT '',
  `dst` VARCHAR(80) NOT NULL DEFAULT '',
  `realsrc` VARCHAR(80) NOT NULL DEFAULT '',
  `realdst` VARCHAR(80) NOT NULL DEFAULT '',
  `dcontext` VARCHAR(80) NOT NULL DEFAULT '',
  `channel` VARCHAR(80) NOT NULL DEFAULT '',
  `dstchannel` VARCHAR(80) NOT NULL DEFAULT '',
  `lastapp` VARCHAR(80) NOT NULL DEFAULT '',
  `lastdata` VARCHAR(80) NOT NULL DEFAULT '',
  `duration` INT(11) NOT NULL DEFAULT '0',
  `billsec` INT(11) NOT NULL DEFAULT '0',
  `disposition` VARCHAR(45) NOT NULL DEFAULT '',
  `amaflags` INT(11) NOT NULL DEFAULT '0',
  `accountcode` VARCHAR(20) NOT NULL DEFAULT '',
  `uniqueid` VARCHAR(32) NOT NULL DEFAULT '',
  `userfield` VARCHAR(255) NOT NULL DEFAULT '',
  `filename` VARCHAR(255) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`),
  KEY `calldate` (`calldate`),
  KEY `accountcode` (`accountcode`),
  KEY `uniqueid` (`uniqueid`),
  KEY `dst` (`dst`),
  KEY `src` (`src`) ) ENGINE=InnoDB AUTO_INCREMENT= DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `cel` (  
  `id` INT(20) NOT NULL AUTO_INCREMENT,  
  `eventtype` VARCHAR(30) NOT NULL,  
  `eventtime` datetime NOT NULL,  
  `cid_name` VARCHAR(80) NOT NULL,  
  `cid_num` VARCHAR(80) NOT NULL,  
  `cid_ani` VARCHAR(80) NOT NULL,  
  `cid_rdnis` VARCHAR(80) NOT NULL,  
  `cid_dnid` VARCHAR(80) NOT NULL,  
  `exten` VARCHAR(80) NOT NULL,  
  `context` VARCHAR(80) NOT NULL,  
  `channame` VARCHAR(80) NOT NULL,  
  `src` VARCHAR(80) DEFAULT NULL,  
  `dst` VARCHAR(80) DEFAULT NULL,  
  `channel` VARCHAR(80) DEFAULT NULL,  
  `dstchannel` VARCHAR(80) DEFAULT NULL,  
  `appname` VARCHAR(80) NOT NULL,  
  `appdata` VARCHAR(80) NOT NULL,  
  `amaflags` INT(11) NOT NULL,  
  `accountcode` VARCHAR(20) NOT NULL,  
  `uniqueid` VARCHAR(32) NOT NULL,  
  `linkedid` VARCHAR(32) NOT NULL,  
  `peer` VARCHAR(80) NOT NULL,  
  `userdeftype` VARCHAR(255) NOT NULL,  
  `eventextra` VARCHAR(255) DEFAULT NULL,  
  `userfield` VARCHAR(255) DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  KEY `uniqueid_index` (`uniqueid`),  
  KEY `linkedid_index` (`linkedid`),  
  KEY `eventtime` (`eventtime`),  
  KEY `exten` (`exten`),  
  KEY `eventtype` (`eventtype`)  
) ENGINE=InnoDB AUTO_INCREMENT= DEFAULT CHARSET=utf8;  
  
CREATE TABLE IF NOT EXISTS `queue_log` (  
  id INT(10) UNSIGNED NOT NULL AUTO_INCREMENT,  
  `time` TIMESTAMP NULL DEFAULT '0000-00-00 00:00:00',  
  callid VARCHAR(32) NOT NULL DEFAULT '',  
  queuename VARCHAR(32) NOT NULL DEFAULT '',  
  agent VARCHAR(32) NOT NULL DEFAULT '',  
  event VARCHAR(32) NOT NULL DEFAULT '',  
  data1 VARCHAR(100) NOT NULL DEFAULT '',  
  data2 VARCHAR(100) NOT NULL DEFAULT '',  
  data3 VARCHAR(100) NOT NULL DEFAULT '',  
  data4 VARCHAR(100) NOT NULL DEFAULT '',  
  data5 VARCHAR(100) NOT NULL DEFAULT '',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

2. Инсталлируем необходимые пакеты (для примера в Debian/Ubuntu)

```
aptitude install unixodbc-dev libmyodbc
```

3. Asterisk должна быть собрана со следующими опциями

```

*****
Asterisk Module and Build Option Selection
*****

Press 'h' for help.

Add-ons (See README-addons.txt)
Applications
Bridging Modules
---> Call Detail Recording
Channel Event Logging
Channel Drivers
Codec Translators
Format Interpreters
Dialplan Functions
PBX Modules
Resource Modules
Test Modules
Compiler Flags
Voicemail Build Options
Utilities
AGI Samples
Module Embedding
Core Sound Packages
Music On Hold File Packages
Extras Sound Packages

*****
Asterisk Module and Build Option Selection
*****

Press 'h' for help.

[ ] --- core ---
[*] cdr_adaptive_odbc
[*] cdr_custom
[*] cdr_manager
[*] cdr_syslog
--- extended ---
[*] cdr_csv
[*] cdr_odbc
XXX cdr_pgsql
XXX cdr_radius
[*] cdr_sqlite3_custom
XXX cdr_tds
--- deprecated ---
XXX cdr_sqlite

```

4. Далее редактируем несколько конфиг-файлов:

/etc/asterisk/res_odbc.conf

```

[asteriskcdrdb]
enabled => yes
dsn => MySQL-asterisk
username => asterisk_user
password => 232d2edxse3e

```

/etc/asterisk/cdr_adaptive_odbc.conf

```
[cdr_adaptive_connection]
connection=asterisk
table=cdr
alias start => calldate
```

/etc/odbc.ini

```
[MySQL-asterisk]
Description = MySQL Asterisk database
;Trace = Off
;TraceFile = stderr
Driver = MySQL
Server = localhost
User = asterisk_user
Password = 232d2edxse3e
;Port = 3306
Socket = /var/run/mysqld/mysqld.sock
Database = asteriskcdrdb
Charset = utf8
```

/etc/odbcinst.ini

```
[MySQL]
Description = MySQL driver
Driver = /usr/lib/x86_64-linux-gnu/odbc/libmyodbc.so
Setup = /usr/lib/x86_64-linux-gnu/odbc/libodbcmyS.so
CPOutput =
CPReuse =
```

В конец **/etc/asterisk/cdr_mysql.conf** добавить

```
alias filename => filename
```

5. Внесем в **/etc/asterisk/extconfig.conf**

```
queue_log => odbc,asteriskcdrdb
```

6. **cel.conf** выглядит так

```
[general]
enable=yes
apps=all
events=all
dateformat = %F %T
[manager]
[radius]
```

7. **cel_odbc.conf**

```
[asteriskcdrdb]
connection=asteriskcdrdb
```

```
table=cel
loguniqueid=yes
charset=utf8
```

8. Проверим все ли в порядке с подключением Asterisk к MySQL

```
*CLI> odbc show asterisk
ODBC DSN Settings
-----
Name:    asterisk
DSN:     MySQL-asterisk
Last connection attempt: 1970-01-01 07:00:00
Pooled:  No
Connected: Yes
```

9. И последним (далеко не в плане значимости) шагом будет настройка диалплана для записи разговоров. Ниже приведен пример макроса для синтаксиса AEL (**extensions.ael**)

```
globals {
    WAV=/var/calls; //Временный каталог с WAV
    MP3=/var/calls; //Куда выгружать mp3 файлы
    RECORDING=1; // Запись, 1 - включена.
};

macro recording (calling,called) {
    if ("${RECORDING}" = "1"){
        Set(CUTTEDCHANNEL=${CUT(CHANNEL,,1)});
        Set(CUTTEDCHANNEL=${CUT(CUTTEDCHANNEL,,2)});
        Set(fname=${UNIQUEID}-${STRFTIME(${EPOCH},,%Y-%m-%d-%H_%M)}-${calling}-${called});
        Set(datedir=${STRFTIME(${EPOCH},,%Y/%m/%d)});
        System(mkdir -p ${MP3}/${datedir});
        System(mkdir -p ${WAV}/${datedir});
        Set(monopt=nice -n 19 /usr/bin/lame -b 32 --silent
"${WAV}/${datedir}/${fname}.wav" "${MP3}/${datedir}/${fname}.mp3" &&
rm -f "${WAV}/${fname}.wav" && chmod o+r
"${MP3}/${datedir}/${fname}.mp3");
        Set(CDR(filename)=${fname}.mp3);
        Set(CDR(realdst)=${called});
        Set(CDR(realsrc)=${calling});
        ExecIf("${calling}" =
"${called}")?Set(CDR(realsrc)=${CUTTEDCHANNEL});
        MixMonitor(${WAV}/${datedir}/${fname}.wav,b,${monopt});
    };
};

_XXXXXX => {
    &recording(${CALLERID(number)},{EXTEN});
    Dial(SIP/83843${EXTEN}@multifon,180,tT);
    HangUP();
};
```

```
} // end of _XXXXXX
```

В данном примере файлы записей разговоров попадают прямыми в **/var/calls**, где имеют следующую иерархию

```
ls /var/calls/2016/ -l
total 24
drwxr-xr-x 19 asterisk asterisk 4096 May 31 10:10 05
drwxr-xr-x 30 asterisk asterisk 4096 Jun 30 10:02 06
drwxr-xr-x 31 asterisk asterisk 4096 Jul 31 10:18 07
drwxr-xr-x 31 asterisk asterisk 4096 Aug 31 09:00 08
drwxr-xr-x 26 asterisk asterisk 4096 Sep 26 09:51 09
```

10. Если вам ближе стандартный синтаксис, обратите внимание на следующий макрос

```
[globals]
WAV=/records
MP3=/records/mp3
FETGRECORDING=1
[macro-fetg.uz-record]
exten => s,1,NoOp(== START macro-fetg.uz-record ==)
    same => n,Set(CallerID=${ARG1})
    same => n,Set(CalleeID=${ARG2})
    same => n,Set(CUTTEDCHANNEL=${CUT(CHANNEL,,1)})
    same => n,Set(CUTTEDCHANNEL=${CUT(CUTTEDCHANNEL,/,2)})
    same => n,NoOp(CalleeID=${CalleeID}, CallerID=${CallerID})
    same => n,NoOp(FETGRECORDING = ${FETGRECORDING})
    same => n,GotoIf(["${FETGRECORDING}" = "0"]?end)
    same => n,Set(fname=${UNIQUEID}-${STRFTIME(${EPOCH},,%Y-%m-%d-%H_%M)}-${
    ${CallerID}-${CalleeID}})
    same => n,Set(datedir=${STRFTIME(${EPOCH},,%Y/%m/%d)})
    same => n,System(mkdir -p ${MP3}/${datedir})
    same => n,Set(monopt=nice -n 19 /usr/bin/lame -b 32 --silent
"${WAV}/${fname}.wav" "${MP3}/${datedir}/${fname}.mp3" && rm -f
"${WAV}/${fname}.wav" && chmod o+r "${MP3}/${datedir}/${fname}.mp3")
    same => n,Set(CDR(filename)=${fname}.mp3)
    same => n,Set(CDR(realdst)=${CalleeID})
    same => n,Set(CDR(realsrc)=${CallerID})
    same => n,ExecIf(["${CallerID}" =
"${CalleeID}"]?Set(CDR(realsrc)=${CUTTEDCHANNEL}))
    same => n,MixMonitor(${WAV}/${fname}.wav,b,${monopt})
    same => n(end),MacroExit()

;пример использования
[from-internal-custom]
;include => from-internal
exten => _XX.,1, Macro(fetg.uz-record,${CALLERID(number)},{EXTEN})

[from-pstn-custom]
```

```
exten => _XX.,1, Macro(fetg.uz-record,${CALLERID(number)},{EXTEN})
```

11. Или у вас внезапно lua

```
WAV = "/opt/records/wav" -- Временный каталог с WAV
MP3="/opt/records/mp3" --Куда выгружать mp3 файлы
RECORDING=1 -- 1 - запись включена.

function recording(calling,called)
    if RECORDING == 1 then
        fname = channel.UNIQUEID:get() .. "-" ..
channel.STRFTIME("${EPOCH},,%Y-%m-%d-%H_%M".."-"..calling.."-"
"..called):get()
        datedir = channel.STRFTIME("${EPOCH},,%Y/%m/%d"):get().."/"
        os.execute("mkdir -p " .. WAV.."/"..datedir)
        os.execute("mkdir -p " .. MP3.."/"..datedir)
        WAVfname = WAV..datedir..fname.."wav"
        MP3fname = MP3..datedir..fname.."mp3"
        channel.CDR("filename"):set(MP3fname)
        channel.CDR("realdst"):set(called)
        channel.CDR("realsrc"):set(calling)
        monopt = string.format("nice -n 19 /usr/bin/lame -b 32 --silent %s
%s && rm -f %s && chown o+r %s",WAVfname,MP3fname,WAVfname,MP3fname)
        app.mixmonitor(WAVfname,b,monopt)
        return
    end;
end;

extensions = {
    ["local"] = {
        --локальные вызовы на номера XXX
        ["_XXX"] = function(context,extension)
            app.answer()
            recording(channel.CALLERID("num"):get(),extension)
            app.dial("SIP/"..extension,300,"t")
            app.hangup()
        end;
    };
};
}
```

12. С препарированием нашей АТС закончили!

Переходим к настройке синхронизации БД и файлов записей разговоров в облако FETG.UZ. Для этого воспользуйтесь инструкцией - https://wiki.fetg.uz/doku.php?id=cloud_daemon_sync

13. Для входа в систему статистики используйте данные авторизации (Email-адрес/Пароль) из регистрационного письма.

На этом все настройки завершены.

Если вы все сделали правильно, то через какое-то время (зависит от размера БД) сможете воспользоваться всеми отчетами системы статистики FETG.UZ в своем личном кабинете. Если у вас остались вопросы и/или есть предложения/замечания, пишите нам на

support@fetg.uz.

From:

<https://wiki.fetg.uz/> - Система статистики call центров на IP-ATC Asterisk (FreePBX)

Permanent link:

https://wiki.fetg.uz/doku.php?id=configure_asterisk_for_cloud_version



Last update: **2022/10/14 07:13**