

Stargazer — руководство пользователя

Copyright © 2016 Maxim Mamontov

COLLABORATORS

	<i>TITLE :</i> Stargazer — руководство пользова- теля		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Maxim Mamontov	22 декабря 2016 г.	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Оглавление

1 Назначение и возможности системы.	1
2 Описание компонентов системы.	2
3 Установка системы.	3
3.1 Установка сервера	3
3.2 Установка авторизатора и конфигуратора	5
3.3 Запуск сервера	6
4 Настройка системы	7
4.1 Конфигурационный файл /etc/stargazer/stargazer.conf	7
4.2 Описание параметров модуля InetAccess (auth_ia) для работы с авторизаторами абонентов:	10
4.3 Модуль авторизации auth_ao.	11
4.4 Описание параметров модуля SGConfig (conf_sg) для работы с конфигуратором: . . .	11
4.5 Описание параметров модуля файловой БД:	11
4.6 Описание параметров модуля для работы с СУБД Firebird:	12
4.7 Описание параметров модуля для работы с СУБД PostgreSQL:	12
4.8 Описание параметров модуля для работы с СУБД MySQL:	13
4.9 Описание параметров модуля ping для пингования абонентов:	13
4.10 Описание параметров модуля Remote Script Executer (remote_script) для передачи команд на исполнение скриптов на NAS:	13
4.11 Описание параметров модуля radius для поддержки авторизации и аккаунтинга пользователей через сервер FreeRADIUS:	14
4.12 Описание параметров модуля XML-RPC (conf_rpc) для поддержки протокола управления XML-RPC:	14
4.13 Модуль захвата трафика cap_ether (только для ОС Linux).	14
4.14 Модуль захвата трафика cap_ipq (только для ОС Linux).	15
4.15 Описание параметров модуля cap_bpf для захвата трафика (только для ОС FreeBSD):	15
4.16 Модуль захвата трафика cap_divert (только для ОС FreeBSD).	15
4.17 Описание параметров модуля cap_nf для захвата трафика:	15
4.18 Описание параметров модуля smux для мониторинга состояния Stargazer:	16
4.19 Конфигурационный файл /etc/stargazer/rules	19
4.20 Использование скриптов OnConnect, OnDisconnect, OnUserAdd, OnUserDel, OnChange	20
4.21 Описание конфигурационного файла stg.conf для плагина FreeRADIUS rlm_stg.so . .	21

5	Описание работы с конфигуратором	22
6	Описание работы с авторизатором InetAccess.	40
6.1	Описание настроек авторизатора.	44
7	Описание работы с консольным авторизатором	47
8	Описание работы с консольным конфигуратором	48
9	Лицензия, авторы, копирайты, благодарности.	51
10	Приложение А: конвертация БД	52
11	Приложение Б	54
11.1	Описание формата файлов /var/stargazer/admins/*.adm.	54
11.2	Описание формата файлов /var/stargazer/users/*/*.	54
11.3	Описание формата файлов /var/stargazer/tariff/*.tf.	56
12	Приложение В	58
12.1	Описание процесса ручной установки БД для СУБД Firebird	58
12.2	Описание структуры БД для СУБД Firebird	58
13	Приложение Г	65
13.1	Описание процесса установки БД для СУБД MySQL	65
13.2	Описание структуры БД для СУБД MySQL	66

Глава 1

Назначение и возможности системы.

Система Stargazer (далее просто «система») предназначена для авторизации и учета трафика в локальных сетях. При написании данной системы была поставлена задача создать продукт, который отвечал бы требованиям большинства локальных сетей для учета в них трафика и средств клиентов, а также безопасной авторизации клиентов.

Система построена по клиент-серверной технологии, что обеспечивает необходимую гибкость и быстродействие. В качестве сервера выступает машина с ОС Linux или FreeBSD и архитектурой x86/x86_64/SPARC. В качестве клиентов могут выступать машины как семейства Windows, так и клиенты с ОС Linux или FreeBSD. Так же клиентом может выступать любая ОС, в которой есть поддержка сетевых протоколов TCP-IP и WEB-браузер.

Система имеет модульную структуру, что позволяет гибко изменять ее конфигурацию и функциональность.

Основные возможности системы:

- контроль над клиентами сети, их добавление, удаление, текущие корректировки;
- авторизация клиента, с последующим разрешением или запретом доступа в Internet;
- подсчет трафика по предварительно заданным направлениям и правилам;
- подсчет израсходованных клиентом средств и автоматическое отключение в случае их полного расходования;
- хранение дополнительной информации о клиенте, такой как домашний адрес, телефон и т.д.;
- автоматический пинг всех клиентов сети и вывод результатов на экран;
- ведение истории для всех клиентов;
- журналирование действий администраторов системы;
- оперативное предоставление клиенту информации о его трафике и наличии средств;
- формирование подробных отчетов о состоянии клиентов;
- подключаемые модули;

Глава 2

Описание компонентов системы.

Составными компонентами системы являются:

1. Сервер;
2. Конфигуратор;
3. Авторизатор.

Сервер – служит для авторизации клиентов, манипуляции правилами сетевого фильтра (firewall), подсчета трафика, выполнения финансовых операций, хранения информации о статистике и выдаче её клиентам и администраторам.

Конфигуратор – используют администраторы сети. Служит для просмотра и редактирования свойств клиентов. Конфигуратор является основным инструментарием для администрирования системы. Доступ к различным функциям разграничен для различных категорий администраторов. С помощью конфигуратора выполняются все финансовые операции с клиентскими счетами, так же просматривается подробная статистика клиентов. Конфигуратором настраиваются некоторые параметры сервера, например тарифные планы и администраторы.

Авторизатор – используется на клиентских машинах. С его помощью происходит авторизация клиента на сервере, получение права на доступ в Интернет или к другим ресурсам, которые разрешены администратором. Так же авторизатор позволяет клиенту просмотреть свою статистику и состояние денежных средств на счету.

Глава 3

Установка системы.

Система обычно поставляется в виде 3 файлов: исходных кодов сервера, бинарной сборки конфигуратора и авторизатора.

Стабильная версия сервера маркируется так: stg-A.B.src.tgz. А и В – старшая и младшая версия сервера. Промежуточные версии сервера имеют маркировку: stg-A.B-YYYY.MM.DD-НН.ММ.СС.tgz. А и В – старшая и младшая версия сервера. YYYY.MM.DD-НН.ММ.СС – дата и время сборки.

Установочный файл конфигуратора имеет маркировку: sgconfig-A.B.C.win.exe. А.В – старшая и младшая версия конфигуратора. С – версия протокола конфигуратора. Исходные коды конфигуратора имеют маркировку: sgconfig-A.B.C.src.rar.

Установочный файл авторизатора имеет маркировку: inetaccess-A.B.C.win.exe. А.В – старшая и младшая версия авторизатора. С – версия протокола авторизатора.

3.1 Установка сервера

```
tar -zxvf stg-A.B.src.tgz
cd stg-A.B.src/projects/stargazer
./build
make install
```

Заголовок скрипта build содержит набор параметров сборки и установки:

- LDFLAGS – дополнительные ключи компоновщика (ключи, предназначенные непосредственно ld указывать в виде -Wl,<ключ>);
- CFLAGS, CXXFLAGS – дополнительные ключи компилятора;
- PREFIX – префикс путей установки;
- BIN_MODE – права доступа к бинарным файлам системы (0755 по умолчанию);
- DATA_MODE – права доступа к файлам данных системы (0644 по умолчанию);
- DIR_MODE — права доступа к каталогам системы (0755 по умолчанию);
- OWNER – владелец файлов системы (root по умолчанию);

Параметры указываются в формате <ИМЯ_ПАРАМЕТРА> = <ЗНАЧЕНИЕ_ПАРАМЕТРА>. CFLAGS, CXXFLAGS и LDFLAGS можно передавать в скрипт build из окружения. Например так:

```
CFLAGS=-O2 CXXFLAGS=-O2 LDFLAGS=-Wl,--as-needed ./build
```


Следует учитывать тот факт что исходный код сервера содержит в себе модули написанные как на C++ так и на C, по этому необходимо передавать флаги компилятора через обе переменные: CFLAGS и CXXFLAGS. При сборке сервера для использования на «боевом» сервере (или для включения в дистрибутив) настоятельно рекомендуется использовать оптимизацию (флаг компилятора -O). Обычно достаточно указать -O2. При отладочной сборке, в общем случае, оптимизацию лучше не включать.

В случае успешной сборки бинарных файлов сервера и их инсталляции в каталогах /etc и /var появится каталог stargazer с набором конфигурационных файлов. Бинарный файл установится в каталог /usr/sbin с именем stargazer, библиотеки и модули установятся в каталог /usr/lib/stg/ а заголовочные файлы в каталог /usr/include/stg.

Если сборка прошла без ошибок и необходимые файлы находятся в соответствующих каталогах, то установка считается успешно завершённой.

Для удобства разработчиков и для создания багрепортов возможна сборка в отладочном режиме. Для этого скрипт ./build запускается с параметром debug. В этом режиме сборка не производится, а выполняется только конфигурирование. Для сборки отладочной версии необходимо выполнить команду make (gmake для BSD-систем). Также возможна отдельная установка бинарных файлов и файлов данных. Для этого в Makefile предусмотрены цели install-bin и install-data. make install-bin устанавливает только бинарные файлы (сервер, модули и библиотеки) и заголовочные файлы. make install-data устанавливает конфигурационные файлы и файловую БД. По аналогии с целью install работает и цель uninstall. make uninstall удаляет сервер из системы. uninstall-bin и uninstall-data удаляют бинарные файлы и файлы данных соответственно.

При обновлении системы настоятельно рекомендуется использовать команду make install-bin.

Внимание! Для успешной сборки сервера необходима правильно установленная библиотека LibExpat. (Поиск производится в /usr/lib и /usr/local/lib, но при желании с помощью переменных окружения CFLAGS/CXXFLAGS и LDFLAGS можно указать и иное месторасположение библиотеки).

Если в процессе сборки будет обнаружена правильно установленная библиотека fbcclient (входит в стандартную поставку СУБД Firebird) в сборку будет включен модуль для работы с СУБД Firebird. Модуль поддерживает работу с СУБД Firebird версии 1.5 и выше.

Если в процессе сборки будет обнаружена правильно установленная библиотека mysqlclient (входит в стандартную поставку СУБД MySQL) в сборку будет включен модуль для работы с СУБД MySQL. Модуль поддерживает работу с СУБД MySQL версии 4.0 и выше.

Если в процессе сборки будет обнаружена библиотека pq в сборку будет включен модуль для работы с СУБД PostgreSQL. Модуль поддерживает работу с СУБД PostgreSQL версии 8.0 и выше.

Примечание: Если сборка завершилась неудачей, то необходимо проверить доступность библиотеки LibExpat. Пользователи ОС Linux могут выполнить команду `ldconfig -p | grep libexpat.so`. Если она выведет нечто похожее на

```
# ldconfig -p | grep libexpat.so
libexpat.so.1 (libc6) => /usr/lib/libexpat.so.1
libexpat.so (libc6) => /usr/lib/libexpat.so
```

значит библиотека доступна. Если вывод будет пустой – библиотека в системе не найдена. Для пользователей ОС FreeBSD можно выполнить команду `ls /usr/lib | grep libexpat.so`. Если она выведет нечто похожее на

```
$ ls /usr/lib | grep libexpat.so
libexpat.so
libexpat.so.4
```

значит библиотека доступна. Если вывод пустой – скорее всего библиотеки в системе нет или она установлена неправильно.

Возможно, библиотека установлена в другое место. Если это так – достаточно при выполнении скрипта build в CFLAGS/CXXFLAGS с помощью ключа -I указать путь к заголовочным файлам

библиотеки, а в LDLFAGS с помощью ключа `-L` указать путь к самой библиотеке. Для поиска библиотеки можно использовать команды `locate`, `slocate`, `whereis` или `find`. Если библиотека в системе отсутствует – установите ее используя Ваш пакетный менеджер (`apt` (Debian, Ubuntu), `rpm` (RPM-based дистрибутивы), `emerge` (Gentoo) или др.). Так же библиотеку можно установить из исходных кодов, доступных на странице <http://sourceforge.net/projects/expat/>.

Для использования СУБД Firebird в качестве хранилища необходимо наличие библиотеки `fbclient`. Проверка может быть выполнена аналогично библиотеке `LibExpat`: `ldconfig -p | grep libfbclient.so` для ОС Linux и `ls /usr/lib | grep libfbclient.so` для ОС FreeBSD. Некоторые дистрибутивы позволяют установить ее отдельно от СУБД (например, в Ubuntu соответствующий пакет называется `libfbclient1`), некоторые – только в составе сервера (например, Gentoo). Исходные коды СУБД Firebird доступны для скачивания со страницы <http://www.firebirdsql.org/index.php?op=files>. Необходимо заметить, что сами разработчики СУБД Firebird не разделяют свой продукт на клиентскую и серверную части. По этому архивы исходных кодов содержат и сервер СУБД, и консольный клиент (командный интерпретатор SQL) и клиентскую библиотеку.

Для использования СУБД MySQL в качестве хранилища необходимо наличие библиотеки `mysqlclient`. Проверка выполняется аналогично первым двум: `ldconfig -p | grep libmysqlclient_r.so` для ОС Linux и `ls /usr/lib | grep libmysqlclient_r.so` для ОС FreeBSD. Так же проверку можно выполнить с помощью утилиты `mysql_config`. Ее вызов с ключем `--libs_r` должен выдать что-то похожее на

```
$ mysql_config --libs
-Wl,-O1 -rdynamic -L/usr/lib/mysql -lmysqlclient -lz -lcrypt -lnsl -lm -L/usr/lib -lssl -lcrypto ↵
```

Необходима именно реентерабельная версия библиотеки, т.к. Сам плагин не имеет блокировок. Некоторые дистрибутивы позволяют установить ее отдельно от СУБД (например, в Ubuntu соответствующие пакеты называются `libmysqlclient15-dev` и `libmysqlclient15off`), некоторые – только в составе сервера (например, Gentoo). Исходные коды открытой/свободной версии СУБД MySQL доступны для скачивания со страницы <http://dev.mysql.com/downloads/mysql/5.0.html>. Необходимо заметить, что сами разработчики СУБД MySQL не разделяют свой продукт на клиентскую и серверную части. По этому архивы исходных кодов содержат и сервер СУБД MySQL, и консольный клиент (командный интерпретатор SQL) и клиентскую библиотеку.

Для использования СУБД PostgreSQL необходимо наличие библиотеки `pq`. Проверка выполняется аналогично предыдущим: `ldconfig -p | grep libpq.so` для ОС Linux и `ls /usr/lib | grep libpq.so` для ОС FreeBSD. Так-же проверку можно выполнить с использованием утилиты `pg_config` или `pkg_config`. Ее вызов с ключем `--libdir` выдать что-то похожее на

```
$ pg_config --libdir
/usr/lib
```

Обычно библиотека устанавливается отдельно от самой СУБД. Например, в Debian соответствующий пакет называется `libpq5` и `libpq-dev`. Исходные коды библиотеки можно загрузить со страницы <http://www.postgresql.org/ftp/source/>. Необходимо заметить, что сами разработчики СУБД PostgreSQL не разделяют свой продукт на клиентскую и серверную части. По этому архивы исходных кодов содержат и сервер СУБД PostgreSQL, и консольный клиент (командный интерпретатор SQL) и клиентскую библиотеку.

3.2 Установка авторизатора и конфигулятора

Для установки конфигулятора необходимо запустить установочный файл и следовать вопросам мастера установки.

Для установки авторизатора необходимо запустить установочный файл и следовать вопросам мастера установки.

3.3 Запуск сервера

При запуске в качестве первого параметра можно указать путь к директории, где расположен файл `stargazer.conf`. По умолчанию при запуске без параметров система ищет файл в каталоге `/etc/stargazer`. Например если файл `stargazer.conf` находится в каталоге `/etc/stargazer2`, то для запуска системы с чтением настройки из этого каталога следует запустить сервер таким образом: `/usr/sbin/stargazer /etc/stargazer2`.

При использовании на “боевом” сервере настоятельно рекомендуется использовать `init.d`-скрипты для запуска, останова, перезапуска и реконфигурирования сервера. Заготовки скриптов для дистрибутивов Ubuntu, Suse и Gentoo входят в стандартную поставку и расположены в каталоге `projects/stargazer/inst/linux/etc/init.d`. Уважайте себя и других людей, которые, возможно, завтра будут администрировать этот сервер.

Для просмотра, работает ли сервер, необходимо выполнить команду: `ps -x | grep stargazer | grep -v grep`. В случае если сервер успешно был запущен и работает то результат команды должен быть примерно таким:

```
4818 ?    S<    0:32 ./stargazer /etc/stargazer
4819 ?    S     0:00 stg-exec   /etc/stargazer
```

Следует отметить что в выводе указанной выше команды должно присутствовать не менее двух строк. То есть, первая строка в данном примере – это процесс сервера, вторая – процесс исполнителя скриптов. Если присутствует только строка с именем `stg-exec` или не имеется строк вовсе, то сервер не запущен или имеются проблемы в его работе. Для диагностики ошибок при запуске сервера необходимо просмотреть содержимое журнала сервера биллинга (`/var/log/stargazer.log`) или же системный журнал (обычно `/var/log/messages`).

Сервер ведет журнал своих действий, куда заносятся все сообщения системы. По умолчанию лог файл находится в каталоге `/var/log` и имеет название `stargazer.log`. Если при запуске сервера возникла какая-то ошибка, то необходимо просмотреть файл журнала и исправить ошибку. В случае если серверу не удастся записать сообщения в указанный файл, он пытается писать сообщения в `syslog`, обычно записи `syslog` находятся в `/var/log/messages`.

Глава 4

Настройка системы

После инсталляции система должна быть подвергнута процедуре настройки. Обычно следует начинать с настройки сервера.

Основные конфигурационные файлы сервера по умолчанию находятся в каталоге `/etc/stargazer`. Они включают в себя: основной конфигурационный файл `stargazer.conf`, файл описания направлений тарификации `rules`, набор скриптов `On*` и два каталога — `conf-available.d` и `conf-enabled.d`, содержащих конфигурационные файлы отдельных модулей. Для включения какого-либо модуля нужно сделать символическую ссылку на его конфигурационный файл в каталоге `conf-enabled.d` или прописать его конфигурационную секцию в разделе `<Modules>` файла `stargazer.conf`.

4.1 Конфигурационный файл `/etc/stargazer/stargazer.conf`

Файл имеет текстовый формат, содержащий пары ПАРАМЕТР = ЗНАЧЕНИЕ и секции `<ИМЯ_СЕКЦИИ ПАРАМЕТРЫ_СЕКЦИИ>`. Комментарии в файле начинаются с символа `#`. В файле описываются общие параметры, которые являются глобальными значениями для всего сервера биллинга, а также параметры соответствующих модулей. Параметры модулей должны быть заключены в секции:

```
<Module имя_модуля>
    список параметров модуля
</Module>
```

Имя модуля представляет собой имя файла этого модуля без префикса `mod_` и суффикса `.so`. Например, секция конфигурации модуля `remote_script` (имя файла: `mod_remote_script.so`) может выглядеть так:

```
<Module remote_script>
    SendPeriod = 15
    SubnetFile = subnets
    Password = 123456
    UserParams = Cash Tariff EnabledDirs
    Port = 9999
</Module>
```

Описание глобальных параметров параметров:

- `Rules` – обязательный параметр, указывающий серверу, где находится файл с правилами для подсчета трафика клиентов. Стандартное значение: `/etc/stargazer/rules`.
- `LogFile` – обязательный параметр, указывающий серверу расположение файла, в который будут записываться все сообщения системы. Стандартное значение: `/var/log/stargazer.log`.

- **PidFile** — необязательный параметр, указывающий серверу где находится файл хранящий PID (идентификатор процесса) демона. По умолчанию это `/var/run/stargazer.pid`. Используется скриптами запуска/останова.
- **ModulesPath** — обязательный параметр, указывающий серверу путь, где находятся подключаемые модули (плагины). Стандартное значение: `/usr/lib/stg`.
- **DetailStatWritePeriod** — обязательный параметр, указывающий серверу через какое время должна записываться детальная статистика клиента. Стандартное значение: `1/6`. Данный параметр может принимать следующие значения: `3` - раз в три часа, `2` - раз в два часа, `1` - раз в час, `1/2` - раз в пол часа, `1/4` - раз в 15 минут, `1/6` - раз в 10 минут.
- **StatWritePeriod** — необязательный параметр, указывающий серверу периодичность записи в БД информации о статистике пользователя в минутах. При большом количестве пользователей эту величину стоит увеличить, так как запись в БД может занимать длительное время. По умолчанию имеет значение `10`.
- **DayFee** — обязательный параметр, указывающий серверу, в какой день месяца снимать с клиентских счетов сумму абонплаты, в соответствии с тарифными планами клиентов. Может принимать значения от `0` до `31`, стандартное значение: `1`. Следует помнить, что в некоторых месяцах не более 28 дней. Значение `0` соответствует последнему дню месяца по календарю.
- **DayResetTraff** — обязательный параметр, указывающий серверу в какой день месяца сбрасывать помесечную статистику клиентов и осуществлять переход на новый тариф, если он был изменен у клиента. Может принимать значения от `0` до `31`, стандартное значение: `1`. Следует помнить, что в некоторых месяцах не более 28 дней. Значение `0` соответствует последнему дню месяца по календарю.
- **SpreadFee** — обязательный параметр, указывающий серверу производить ли «размазанное» снятие абонплаты. Может принимать значения `yes` и `no`. При указании значения `yes` сумма абонплаты указанная в тарифе пользователя будет сниматься не раз в месяц, а каждый день равными долями. Стандартное значение: `no`.
- **FreeMbAllowInet** — обязательный параметр, указывающий серверу выполнять ли для пользователя скрипт `OnConnect`, если у него на балансе отсутствуют средства, но остался prepaid трафик. Может принимать значения `yes` и `no`. Стандартное значение: `no`.
- **WriteFreeMbTraffCost** — обязательный параметр, определяющий как будет писаться стоимость трафика в детальной статистике. Может принимать значения `yes` и `no`, стандартное значение: `no`. При указании значения `yes` в детальной статистике пользователя стоимость трафика будет писаться в любом случае, даже если он попадает в `FreeMb`. При указании значения `no` до истощения `FreeMb` в детальной статистике будет указана нулевая стоимость трафика.
- **DayFeeIsLastDay** — обязательный параметр, указывающий серверу когда снимать абонплату. Может принимать значения `yes` и `no`, стандартное значение: `yes`. Это влияет на то, как будет снята абонплата (АП) при переходе на новый тариф. Если у пользователя был тариф А с АП=100 и он хочет перейти на тариф В с АП=200, то при переходе на новый тариф со счета пользователя снимется 100, если указано значение `yes` (абонплата снята до смены тарифа), и 200, если указано значение `no` (абонплата снята перед сменой тарифа).
- **ExecutorsNum** — необязательный параметр который определяющий сколько скриптом может выполняться одновременно (регулирует количество процессов `stg-exec`). По умолчанию имеет значение `1`. Если выполнение ваших скриптов занимает много времени имеет смысл увеличить это значение. Каждый исполнитель скриптов блокируется на время, пока отработает скрипт. Если исполнителей несколько — они принимают скрипты на выполнение по принципу «кто успел».
- **StgMsgKey** — необязательный параметр отвечающий за идентификатор очереди сообщений для исполнителя скриптов. По умолчанию данный параметр имеет значение `5555`. Рекомендуется не изменять значение параметра без нужды. Его изменение целесообразно лишь в том случае, если в системе требуется запустить более одной копии сервера биллинга.

- **ShowFeeInCash** – необязательный параметр, определяющий как будет показано состояние счета абонента в авторизаторе и сколько денег ему будет разрешено использовать в текущем месяце. Может принимать значения `yes` и `no`, по умолчанию имеет значение `yes`. При указании `no` в авторизаторе будет показано состояние счета без учета следующего снятия абонплаты. Так же эти деньги нельзя будет использовать для оплаты трафика в текущем месяце. Пример: у абонента на счету 200 денег при абонплате в 50 денег — в этом случае ему будет показано 150 денег в авторизаторе и он сможет в текущем месяце израсходовать только 150 денег, но снятие абонплаты не уведет его в «минус».
- **FullFee** – необязательный параметр, указывающий серверу снимать полную абонплату, даже если учетная запись абонента была заморожена (или не существовала) часть учетного периода. Принимает значения `yes` и `no`, по умолчанию установлен в `no`.
- **MessagesTimeout** — необязательный параметр, устанавливающий время жизни не отправленных сообщений абонентам. Время указывается в сутках. При превышении этого времени сообщение будет удалено, в т. ч. из БД. Если указано значение 0 то не отправленные сообщения никогда не будут удаляться из базы (в частности, это приведет к постепенному росту размера базы, увеличению нагрузки на сервер при авторизации абонентов и к тому что долго отсутствовавший абонент при авторизации получит все пропущенные сообщения). По умолчанию имеет значение 0.
- **FeeChargeType** — необязательный параметр, регулирующий процесс снятия абонплаты. Может принимать значения 0, 1 и 2, по умолчанию имеет значение 0. При значении 0 абонплата снимается как обычно, при значении 1 абонплата снимается только если баланс пользователя положительный или равен нулю, при значении 2 абонплата снимается только если баланс пользователя больше или равен абонплате. Значение 2 следует использовать с осторожностью, т. к. при этом на безлимитных тарифах абоненты получают услугу бесплатно.
- **ReconnectOnTariffChange** — необязательный параметр, указывающий серверу выполнить переподключение пользователя при смене тарифа. Может принимать значения `yes` и `no`, по умолчанию имеет значение `no`. При указании значения `yes` подключенные пользователи будут отключены непосредственно перед сменой тарифа и подключены сразу после нее. Может быть полезно для управления шейпером.
- **ScriptParams** — необязательный параметр, который определяет дополнительный набор данных передаваемых в скрипты `OnConnect` и `OnDisconnect`. По умолчанию этот параметр пуст. В нем можно указать названия полей записи пользователя, разделенных пробелами, значения которых будут переданы в скрипты после стандартного набора, включающего `login`, `ip`, `cash`, `id` и `dirs`. Допустимые имена полей:

```
"cash", "upload", "download", "lastCashAdd", "passiveTime", "lastCashAddTime", "freeMb", " ←
  "lastActivityTime", "password", "passive", "disabled", "disabledDetailStat", " ←
  alwaysOnline", "tariffName", "nextTariff", "address", "note", "group", "email", "phone ←
  ", "realName", "credit", "creditExpire", "ips", "userdata0" ... "userdata9".
```

Так, например, если указать `ScriptParams = tariffName userdata0` то список параметров передаваемых в скрипты будет: `login`, `ip`, `cash`, `id`, `dirs`, `tariffName`, `userdata0`. Названия параметров регистронезависимые.

- **DisableSessionLog** — необязательный параметр позволяющий отключать запись сессий пользователя в БД. Может принимать значения `yes` и `no`. По умолчанию установлен в `no`, то есть запись сессий включена.
- **FilterParamsLog** — необязательный параметр который позволяет фильтровать запись в журнал изменений параметров. Может принимать значения `"*"` (разрешена запись журнала для всех параметров) или список названий полей записи пользователя, разделенный пробелами. Допустимые имена полей такие-же как и для `ScriptParams`. По умолчанию имеет значение `"*"`, то есть разрешено журналирование всех параметров. Если указать `FilterParamsLog = cash tariffName password` то в журнал изменений будут попадать только изменения параметров `cash`, `tariffName` и `password`. Изменения других параметров, например `note`, `group` или `address` в журнал не попадут.

- Для именования направлений учета трафика в конфигурационном файле используется секция DirNames:

```
<DirNames>
  DirName0 = «ИМЯ НАПРАВЛЕНИЯ»
  ...
  DirName9 = ПОСЛЕДНЕЕ_НАПРАВЛЕНИЕ
</DirNames>
```

Всего предусмотрено 10 направлений с нумерацией от 0 до 9. Направления, наименование которых состоит из нескольких слов, должны быть заключены в кавычки. Если наименование направления не указано, то система считает что данное направление не используется и не будет производить по нему никаких действий. Пример: DirName0 = Мир, DirName5 = "Локальные игры". Необходимо заметить, что текущая версия системы требует, чтобы имена направлений были указаны в кодировке KOI8-R. В противном случае абоненты не смогут их прочитать в статистике авторизатора, а администраторы — в конфигураторе.

Настройки всех модулей (кроме store module), которые необходимо подключить к серверу в момент старта должны быть указаны внутри секции Modules:

```
<Modules>
  <Module имя_модуля>
    список параметров модуля
  </Module>

  <Module имя_модуля>
    список параметров модуля
  </Module>
</Modules>
```

Если модуль не имеет настраиваемых параметров, то он все равно должен быть указан. Некоторые модули, такие как store module, обязательно требуются при старте и без их подключения система не может быть запущена. По умолчанию для указания конфигурации модулей используются отдельные файлы из каталога conf-enabled.d. Для этого используется директива <IncludeFile ПУТЬ_К_ФАЙЛУ></IncludeFile>. Одна директива указана в секции Modules, а вторая в корне файла конфигурации (для store module).

4.2 Описание параметров модуля InetAccess (auth_ia) для работы с авторизаторами абонентов:

- Port – обязательный параметр, определяющий на каком порту сервер будет принимать обращения авторизаторов абонентов. Стандартное значение: 5555.
- UserTimeout – обязательный параметр, указывающий серверу, через какое время в секундах абонент будет отключен если авторизатор не отвечает серверу на запросы. Может принимать значения от 15 до 1200, стандартное значение: 65.
- UserDelay – обязательный параметр, указывающий серверу время в секундах, через которое происходит опрос авторизатора абонента. Сервер шлет абоненту запрос подтверждения работы авторизатора, а авторизатор абонента должен прислать ответ что он находится в рабочем состоянии. Так же это время, через которое в авторизаторе обновляется статистика. Может принимать значения от 5 до 600, стандартное значение: 60. Следует заметить что значение UserDelay должно быть меньше чем значение UserTimeout.
- FreeMb – обязательный параметр, определяющий в каком виде абоненту будет показано значение FreeMb. Данный параметр может принимать следующие значения:

- 0 - количество бесплатных мегабайт в пересчете на цену нулевого направления;
- 1 - количество бесплатных мегабайт в пересчете на цену первого направления;
- 2 - количество бесплатных мегабайт в пересчете на цену второго направления;
- ...
- 9 - количество бесплатных мегабайт в пересчете на цену девятого направления;
- cash - количество денег на которые клиент может бесплатно работать;
- none - ничего не передавать.

Обмен данными авторизатора с сервером осуществляется по протоколу UDP. Можно указать несколько модулей авторизатора `auth_ia` для авторизации с разных портов.

4.3 Модуль авторизации `auth_ao`.

Модуль параметров не имеет. Используется для поддержки режима Always Online у абонентов. Без включения этого модуля установка параметра `alwaysOnline` для абонента эффекта иметь не будет. В режиме Always Online абонент находится в авторизованном состоянии все время, независимо от использования авторизатора. Тем не менее он может быть отключен по причине отсутствия средств на счету, заблокирован администратором или «заморожен». В этом режиме так же возможно использование авторизатора за одним исключением — абонент не может сам вызвать «отключение».

4.4 Описание параметров модуля `SGConfig (conf_sg)` для работы с конфигуратором:

- `Port` – обязательный параметр, определяющий, на каком порту сервер будет принимать обращения конфигураторов. Стандартное значение: 5555.

Обмен конфигуратора с сервером осуществляется по протоколу TCP. Можно указать несколько модулей конфигуратора с указанием разных портов.

4.5 Описание параметров модуля файловой БД:

- `WorkDir` – обязательный параметр, указывающий серверу где находится рабочая директория с файлами БД. Стандартное значение: `/var/stargazer`.
- `ConfOwner`, `ConfGroup`, `ConfMode` – обязательные параметры, описывающие владельца, группу и права доступа на файлы конфигурации (`conf`) абонента соответственно. `ConfOwner` должен содержать корректное имя пользователя системы (см. файл `/etc/passwd`), стандартное значение: `root`. `ConfGroup` должен содержать корректное название группы в системе (см. файл `/etc/group`), стандартное значение: `root`. `ConfMode` должен содержать корректные права на файл (только `ugo`-биты), стандартное значение: `640`.
- `StatOwner`, `StatGroup`, `StatMode` – обязательные параметры, описывающие владельца, группу и права доступа на файлы статистики (`stat`) абонента соответственно. `StatOwner` должен содержать корректное имя пользователя системы (см. файл `/etc/passwd`), стандартное значение: `root`. `StatGroup` должен содержать корректное название группы в системе (см. файл `/etc/group`), стандартное значение: `root`. `StatMode` должен содержать корректные права на файл (только `ugo`-биты), стандартное значение: `640`.

- UserLogOwner, UserLogGroup, UserLogMode – обязательные параметры, описывающие владельца, группу и права доступа на файлы журналов (log) абонента соответственно. UserLogOwner должен содержать корректное имя пользователя системы (см. файл /etc/passwd), стандартное значение: root. UserLogGroup должен содержать корректное название группы в системе (см. файл /etc/group), стандартное значение: root. UserLogMode должен содержать корректные права на файл (только ugo-биты), стандартное значение: 640.

При создании каталогов (например для записи детальной статистики) используются те-же права, но с добавлением x-бита для всех ненулевых полей. Например: для 640 будут права 750, а для 644 будут 755. Для записи детальной статистики используются параметры StatOwner, StatGroup и StatMode. Для записи сообщений используются параметры ConfOwner, ConfGroup и ConfMode.

4.6 Описание параметров модуля для работы с СУБД Firebird:

- Server – необязательный параметр, описывающий адрес сервера, на котором расположена СУБД. Может быть доменным именем или IP-адресом. Значение по умолчанию: localhost.
- Database – необязательный параметр, описывающий путь к БД на сервере или ее псевдоним (alias). Значение по умолчанию: /var/stg/stargazer.fdb.
- User – необязательный параметр, описывающий имя пользователя БД. Значение по умолчанию: stg.
- Password – необязательный параметр, описывающий пароль пользователя БД. Значение по умолчанию: 123456.
- IsolationLevel — Не обязательный параметр, описывающий уровень изоляции транзакций. Может принимать значения:
 - Concurrency (используется по умолчанию);
 - DirtyRead;
 - ReadCommitted;
 - Consistency.
- LockResolution — Не обязательный параметр, описывающий действие при блокировке. Может принимать значения:
 - Wait (используется по умолчанию);
 - NoWait.

4.7 Описание параметров модуля для работы с СУБД PostgreSQL:

- Server – необязательный параметр, описывающий адрес сервера, на котором расположена СУБД. Может быть доменным именем или IP-адресом. Значение по умолчанию: localhost.
- Database – необязательный параметр, описывающий имя БД. Значение по умолчанию: stargazer.
- User – необязательный параметр, описывающий имя пользователя БД. Значение по умолчанию: stg.
- Password – необязательный параметр, описывающий пароль пользователя БД. Значение по умолчанию: 123456.
- Retries — необязательный параметр, описывающий количество попыток переподключения к СУБД в случае потери связи. Попытки производятся с интервалом в 1 секунду. Значение по умолчанию: 3.

4.8 Описание параметров модуля для работы с СУБД MySQL:

- **Server** – обязательный параметр, описывающий адрес сервера, на котором расположена СУБД. Стандартное значение: localhost.
- **Database** – обязательный параметр, описывающий название БД. Стандартное значение: stg.
- **User** – обязательный параметр, описывающий имя пользователя БД. Стандартное значение: stg.
- **Password** – обязательный параметр, описывающий пароль пользователя БД. Стандартное значение: 123456.

4.9 Описание параметров модуля ping для пингования абонентов:

- **PingDelay** – обязательный параметр, определяющий, время в секундах между пингами одного и того же абонента. Стандартное значение: 15.

4.10 Описание параметров модуля Remote Script Executer (remote_script) для передачи команд на исполнение скриптов на NAS:

- **SendPeriod** – обязательный параметр, определяющий время в секундах между посылками подтверждений того, что клиент находится в состоянии Online. Стандартное значение: 15.
- **Password** – обязательный параметр, определяющий пароль для шифрования обмена между сервером биллинга и клиентом. Стандартное значение: 123456.
- **UserParams** – обязательный параметр, определяющий какие параметры и в каком порядке передаются на NAS. Названия параметров должны быть разделены пробелом. Доступными параметрами являются: Cash, FreeMb, Passive, Disabled, AlwaysOnline, TariffName, NextTariff, Address, Note, Group, Email, RealName, Credit, EnabledDirs, Userdata0...Userdata9. Стандартное значение: Cash Tariff EnabledDirs.
- **Port** – обязательный параметр, определяющий номер порта через который будет происходить обмен данными между сервером биллинга и клиентом. Может принимать значения от 1 до 65535, стандартное значение: 9999.
- **SubnetFile** — обязательный параметр, представляющий собой путь к файлу с описанием соответствия сетей и NAS'ов. Стандартное значение: subnets. При указании относительного пути поиск будет производиться в каталоге с настройками (обычно это /etc/stargazer, но может быть переопределено указанием пути в качестве параметра при старте демона). Файл имеет формат: <сеть в CIDR-нотации> <адрес NAS'a>. Количество сетей не ограничено. Файл перечитывается заново при посылке процессу сигнала SIGHUP. Если файл содержит ошибки при старте демона — система не будет запущена. Если файл содержит ошибки при перечитывании — будут использоваться старые значения.

4.11 Описание параметров модуля radius для поддержки авторизации и аккаунтинга пользователей через сервер FreeRADIUS:

- Port — обязательный параметр, определяющий порт, на который будут приходить запросы от FreeRADIUS. Может принимать значение от 1 до 65535, стандартное значение: 6666.
- Password — обязательный параметр, задающий пароль для шифрования данных, передаваемых на FreeRADIUS. Должен совпадать с паролем в настройке плагина FreeRADIUS (rlm_stg.so). Стандартное значение: 123456.
- AuthServices — необязательный параметр, задающий список сервисов которым будет позволено только авторизация с использованием БД Stargazer. При успешной авторизации абонент не переходит в состояние Online и для него не производится подсчет трафика. Названия сервисов в списке должны разделяться пробелами, по умолчанию этот параметр пуст.
- AcctServices — необязательный параметр, задающий список сервисов, по которым будет производится аккаунтинг. При успешной авторизации в этих сервисах абонент переходит в состояние Online и для него производится подсчет трафика. Необходимо заметить, что в этом случае трафик захватывается как обычно, без использования возможностей протокола RADIUS (через пакет аккаунтинга InterimUpdate), т.к. это не позволяет классифицировать полученный трафик по направлениям. Названия сервисов в списке должны разделяться пробелами, по умолчанию этот параметр пуст.

Обмен данными между плагином и FreeRADIUS происходит по протоколу UDP.

4.12 Описание параметров модуля XML-RPC (conf_rpc) для поддержки протокола управления XML-RPC:

- Port — обязательный параметр, определяющий порт на который будут приходить запросы XML-RPC. Может принимать значения от 1 до 65535, стандартное значение: 8080.
- CookieTimeout — необязательный параметр, задающий время существования авторизационного Cookie в случае отсутствия активности в секундах. Значение по умолчанию: 1800 (30 минут).

4.13 Модуль захвата трафика cap_ether (только для ОС Linux).

Модуль не имеет параметров. Для захвата трафика используются так называемые «raw sockets», которые позволяют получить доступ к Ethernet-фреймам. Перехватывается весь трафик попадающий в сетевую подсистему ядра. При использовании обычной маршрутизации трафик будет посчитан два раза: на входящем интерфейсе и на исходящем. При использовании NAT удвоения трафика не происходит, так как NAT заменяет адрес источника. При интенсивном сетевом обмене или при высокой нагрузке на сервер, на котором происходит захват трафика, модуль может терять отдельные пакеты. Процент потерь тем выше чем выше скорость прохождения пакетов и чем выше загрузка сервера. Модуль рекомендуется использовать для ознакомления или в небольших сетях до 100 абонентов с трафиком до 100 Мбит.

4.14 Модуль захвата трафика `car_ipq` (только для ОС Linux).

Модуль не имеет параметров. Для захвата трафика используются передача пакетов из пространства ядра в пространство пользователя посредством очередей (`ip queue`). Для его работы требуется поддержка `ip queueing` в ядре (модуль `ip_queue.ko`) и специальная настройка файрвола (правило `QUEUE` для `iptables`). Следует обратить внимание на то что обычно требуется два правила в файрволе для полного перехвата: одно для входящих пакетов и одно для исходящих. Модуль гарантирует 100% перехват трафика, но так как пакет перед отправкой обязательно проходит через плагин — может приводить к снижению пропускной способности роутера. При этом следует обратить внимание на нагрузку на процессор, возможно имеет смысл заменить его на более производительный. В противном случае стоит рассмотреть использование модуля `car_nf` для захвата трафика.

4.15 Описание параметров модуля `car_bpf` для захвата трафика (только для ОС FreeBSD):

- `iface` — обязательный параметр, определяющий на каком интерфейсе будет происходить захват трафика. Параметр может быть задан более одного раза.

Для захвата трафика используется инфраструктура Berkeley Packet Filter, представляющая собой «продвинутый» аналог «raw sockets». Так как интерфейсы для перехвата указываются явно, дублирования трафика при обычной маршрутизации не наблюдается. Фильтрация пакетов не используется, перехват происходит по мере возможности, по этому этот модуль, как и `car_ether`, тоже может терять пакеты при высокой нагрузке на сервер или высокой скорости прохождения пакетов. Рекомендуются использовать для ознакомления или для небольших сетей до 100 абонентов с трафиком до 100 Мбит.

4.16 Модуль захвата трафика `car_divert` (только для ОС FreeBSD)

Модуль не имеет параметров. Для захвата трафика используются `divert`-сокеты. Как и `IPQ` эта технология использует прохождение пакетов через пространство пользователя. Для работы модуля требуется поддержка `divert`-сокетов в ядре и специальная настройка файрвола. Для передачи пакета в пространство пользователя в файрволе используется правило `divert` или `tee`. Первое правило работает аналогично цели `QUEUE` для `iptables` — пропускает пакет через пространство пользователя перед отправкой. Соответственно, это может вызвать те-же проблемы с пропускной способностью роутера. Правило `tee` передает в пространство пользователя копию пакета, а оригинал отправляет дальше. Такой подход позволяет избежать снижения пропускной способности сервера при высокой нагрузке на него, так как исключается ожидание пакета в время его нахождения в пространстве пользователя.

4.17 Описание параметров модуля `car_nf` для захвата трафика:

- `TCPPort` — необязательный параметр, указывающий порт который будет использоваться для приема данных при работе с `NetFlow-proxy`. Стандартное значение: 9996. Если параметр не указан — прием по протоколу TCP производиться не будет.
 - `UDPPort` — необязательный параметр, указывающий порт для приема `NetFlow`-даграмм. Стандартное значение: 9996. Если параметр не указан — прием даграмм по протоколу UDP производиться не будет.
-

В явном виде захват трафика не происходит. Данные о нем поступают от NetFlow-сенсора посредством протокола NetFlow. Это позволяет физически разделить перехват трафика и его учет. NetFlow-сенсор перехватывает трафик, и собирает данные о сессиях (в контексте UDP это передача данных между двумя портами в одном направлении). Информация о сессии включает в себя IP-адреса источника и назначения потока пакетов, номера портов источника и назначения, суммарную длину пакетов и различные дополнительные данные. NetFlow-трафик существенно меньше трафика, который он описывает, так как передается мета-информация а не сами данные. С учетом этого факта и того что биллинговый сервер теперь может не заниматься маршрутизацией пакетов и NAT'ом это существенно снижает нагрузку на него. Этот плагин рекомендуется использоваться в крупных сетях с развитой топологией, включающей несколько NAS'ов. Возможно использование совместно с аппаратными маршрутизаторами Cisco (они единственные имеют лицензию на аппаратную реализацию NetFlow). В качестве NetFlow-сенсоров можно использовать такие утилиты как `fprobe`, `softflowd` или `ipcad`. Так же можно использовать модуль ядра `ipt_netflow` (Linux) или `ng_netflow` (FreeBSD).

- `Server` — обязательный параметр, указывающий IP-адрес сервера на котором находится `snmpd`. Стандартное значение: 127.0.0.1.
- `Port` — обязательный параметр, указывающий порт на сервере через который будет происходить взаимодействие с `snmpd`. Стандартное значение: 199.
- `Password` — необязательный параметр, задающий пароль для авторизации плагина в `snmpd`. По умолчанию пароль не используется.

4.18 Описание параметров модуля `smux` для мониторинга состояния Stargazer:

Модуль позволяет производить мониторинг биллинга средствами протокола SNMP. Он не реализует полноценный SNMP-сервер а лишь взаимодействует с существующим демоном `snmpd`, регистрируясь в нем для обслуживания определенного дерева параметров. В комплекте с биллингом идет MIB, описывающий доступные параметры для мониторинга. Параметры разделены на 6 секций, находящихся в узле `stg24`:

- `users (1)` — секция, содержащая параметры относящиеся к абонентам:
 - `totalUsers (1)` — параметр, описывающий общее количество абонентов в системе;
 - `onlineUsers (2)` — параметр, описывающий количество подключенных абонентов;
 - `authorizedUsers (3)` — параметр, описывающий количество авторизованных абонентов в системе;
 - `alwaysOnlineUsers (4)` — параметр, описывающий количество абонентов работающих в режиме `always online`;
 - `noCashUsers (5)` — параметр, описывающий количество абонентов с нулевым или отрицательным балансом;
 - `disabledDetailStatsUsers (6)` — параметр, описывающий количество абонентов с отключенным учетом детальной статистики;
 - `disabledUsers (7)` — параметр, описывающий количество принудительно отключенных абонентов;
 - `passiveUsers (8)` — параметр, описывающий количество «замороженных» абонентов;
 - `creditUsers (9)` — параметр, описывающий количество абонентов с положительным кредитом;
 - `freeMbUsers (10)` — параметр, описывающий количество абонентов с неизрасходованными «бесплатными мегабайтами»;

- tariffChangeUsers (11) — параметр, описывающий количество абонентов у которых с началом месяца будет изменен тариф;
- activeUsers (12) — параметр, описывающий количество абонентов у которых имеется посчитанный трафик за сессию.
- tariffs (2) — секция, содержащая параметры относящиеся к тарифам:
 - totalTariffs (1) — параметр, описывающий общее количество тарифов;
 - tariffUsageTable (2) — таблица использования тарифов, состоит из записей вида:
 - * tariffIndex (1) — числовой индекс в таблице;
 - * tariffName (2) — название тарифа (строка);
 - * userCount (3) — количество абонентов использующих этот тариф.
- admins (3) — секция, содержащая параметры относящиеся к администраторам системы:
 - totalAdmins (1) — параметр, описывающий общее количество администраторов, зарегистрированных в системе.
- services (4) — секция, содержащая параметры относящиеся к услугам:
 - totalServices (1) — параметр, описывающий общее количество услуг по которым ведется учет в системе.
- traffcounter (6) — секция, содержащая параметры относящиеся к учету трафика:
 - totalRules (1) — параметр, описывающий общее количество правил классификации трафика.

К параметрам можно обращаться по цифровым OID'ам или по именам. Для обращения к параметрам по именам нужно предоставить системе файл STG-MIB.mib содержащий информацию об именах, типах и OID'ах параметров с кратким описанием. Этот файл можно поместить в каталог ~/.snmp/mibs (в этом случае он будет доступен лишь одному пользователю) или в каталог /usr/share/snmp/mibs (в этом случае он будет доступен всем пользователям системы). Так же большинство утилит позволяют загружать mib-файлы из произвольного места явным указанием пути к ним. Для доступа к параметрам можно использовать утилиты из пакета NetSNMP (<http://net-snmp.sourceforge.net/>). Для того чтобы просмотреть иерархию параметров можно выполнить такую команду: `snmptranslate -m +STG-MIB -Tp -IR stg24`. Вывод команды должен быть приблизительно таким:

```

+--stg24(1)
|
+--users(1)
| |
| +-- -R-- Integer32 totalUsers(1)
| +-- -R-- Integer32 onlineUsers(2)
| +-- -R-- Integer32 authorizedUsers(3)
| +-- -R-- Integer32 alwaysOnlineUsers(4)
| +-- -R-- Integer32 noCashUsers(5)
| +-- -R-- Integer32 disabledDetailStatsUsers(6)
| +-- -R-- Integer32 disabledUsers(7)
| +-- -R-- Integer32 passiveUsers(8)
| +-- -R-- Integer32 creditUsers(9)
| +-- -R-- Integer32 freeMbUsers(10)
| +-- -R-- Integer32 tariffChangeUsers(11)
| +-- -R-- Integer32 activeUsers(12)
|
+--tariffs(2)
| |
| +-- -R-- Integer32 totalTariffs(1)
| |
| +--tariffUsageTable(2)
| |
| |

```

```

|      +--tariffUsageTableEntry(1)
|      |      Index: tariffIndex
|      |
|      |      +-- ---- Integer32 tariffIndex(1)
|      |      |      Range: 0..255
|      |      +-- -R-- String    tariffName(2)
|      |      |      Textual Convention: DisplayString
|      |      |      Size: 0..255
|      |      +-- -R-- Integer32 userCount(3)
|
|  +--admins(3)
|  |
|  |  +-- -R-- Integer32 totalAdmins(1)
|  |
|  +--services(4)
|  |
|  |  +-- -R-- Integer32 totalServices(1)
|  |
|  +--corporations(5)
|  |
|  |  +-- -R-- Integer32 totalCorporations(1)
|  |
|  +--traffcounter(6)
|  |
|  |  +-- -R-- Integer32 totalRules(1)

```

Если в качестве ответа будет строка «Unknown object identifier: stg24» то это означает что команда не смогла найти mib-файл. Для исправления этой ошибки следует либо скопировать файл STG-MIB.mib по одному из указанных выше путей или указать путь к каталогу где он находится с помощью ключа -M. Символ «+» перед STG-MIB обозначает что данные этого файла должны использоваться совместно с загружаемыми по умолчанию. Без этого символа данные будут использоваться автономно. Ключ -M работает схожим образом: символ «+» говорит о том что указанный путь нужно добавить к путям поиска mib-файлов, используемым по умолчанию. Без символа «+» будет использоваться единственный путь — указанный.

Для доступа к значениями параметров можно использовать следующую команду: `snmpget -v2c -ccommunity_w -m +STG-MIB 10.0.0.1 stg24.users.totalUsers`. Ключ -v указывает используемую версию протокола. На сегодняшний день существует 3 версии протокола: 1, 2 и 3. Версия 1 считается устаревшей и почти не используется. Версия 2 существует в двух модификациях: user-based (с суффиксом «u») и community-based (с суффиксом «c»). Версия 3 является самой новой и предоставляет широкие средства аутентификации, контроля целостности и шифрования. Ключ -c задает community для доступа к серверу. Ключ -m позволяет подключать дополнительные mib-файлы. При использовании доступа к параметрам по названию необходимо подключить mib-файл STG-MIB. Далее следуют два аргумента: адрес SNMP-сервера и название параметра. Утилита `snmpget` позволяет получить значение скалярных параметров — параметров имеющих только одно значение. Для доступа к таблицам и деревьям используется утилита `snmpwalk` имеющая такой-же синтаксис.

Можно обращаться к параметрам по цифровому OID: `snmpwalk -v2c -ccommunity_w 10.0.0.1 .1.3.6.1.4.1.38313.1.2`. В этом случае загружать дополнительные mib-файлы не требуется. «.1.3.6.1.4.1» - OID enterprise-ветки всего дерева SNMP. «38313» - официально полученный от IANA (<http://www.iana.org>) enterprise-номер, уникально идентифицирующий дерево параметров Stargazer. Следующая за ним цифра 1 говорит о том что мы работаем с узлом stg24. «.1.3.6.1.4.1.38313.1.2» - OID таблицы tariffUsageTable.

Для того чтобы установить взаимодействие между плагином `smux` и демоном `snmpd` нужно провести дополнительную настройку последнего. А именно: указать в конфигурационном файле (обычно это `/etc/snmp/snmpd.conf`) параметр `smuxpeer` (OID обслуживаемый плагином, в нашем случае это `.1.3.6.1.4.1.38313`) и `smuxsocket` (IP-адрес с которого будут приходить пакеты от `smux`-плагина).

4.19 Конфигурационный файл /etc/stargazer/rules

Файл rules описывает правила классификации трафика по направлениям тарификации. Это текстовый файл, каждая строка которого описывает одно правило классификации. Формат строки файла:

```
<протокол> <CIDR>[:<порт>[-<порт>]] <направление>
```

Допустимы следующие варианты указания протокола:

- TCP — под правило будет попадать только TCP-трафик;
- UDP — под правило будет попадать только UDP-трафик;
- ICMP — под правило будет попадать только ICMP-трафик;
- TCP_UDP — под правило будет попадать только TCP- и UDP-трафик;
- ALL — под правило будет попадать любой трафик.

Варианты указания фильтра по IP-адресу и портам:

- 192.168.0.1 — под правило будет попадать только трафик хоста 192.168.0.1;
- 192.168.0.1/32 — правило действует аналогично предыдущему;
- 192.168.0.0/24 — под правило будет попадать только трафик сети 192.168.0.0/24;
- 192.168.0.1:80 — под правило будет попадать только web-трафик хоста 192.168.0.1;
- 192.168.0.0/24:80 — под правило будет попадать только web-трафик сети 192.168.0.0/24;
- 192.168.0.1:1024-65535 — под правило будет попадать только трафик с высоких номеров портов хоста 192.168.0.1;
- 192.168.0.1/24:1024-65535 — под правило будет попадать только трафик с высоких номеров портов сети 192.168.0.0/24.

В качестве направления указывается один из идентификаторов DIR0-DIR9 или NULL. Трафик классифицируемый по направлению NULL системой никак не учитывается.

В файле допустимо наличие пустых строк и комментариев. Комментарием считается строка начинающаяся с символа #. Текст следующий за этим символом до конца строки не учитывается.

Классификация трафика происходит путем последовательного сравнения информации из заголовка пакета с правилами в файле сверху-вниз до первого совпадения. По этому более «узкие» правила должны описываться перед более «широкими». Например, чтобы выделить web-трафик определенного хоста в отдельное направление порядок указания правил должен быть следующим:

```
TCP 192.168.0.1:80 DIR1  
ALL 192.168.0.1 DIR2
```

Если поменять эти строки местами то весь трафик хоста 192.168.0.1 попадет под первое правило.

4.20 Использование скриптов OnConnect, OnDisconnect, OnUserAdd, OnUserDel, OnChange

Для реакции на различные события в системе используются файлы OnConnect, OnDisconnect, OnUserAdd, OnUserDel, OnChange. Эти файлы расположены в каталоге с конфигурационными файлами системы и должны иметь бит выполнения. Обычно это скрипты на одном из скриптовых языков программирования (sh, perl, python, php), но могут быть и полноценными программами.

Наиболее часто используются файлы OnConnect и OnDisconnect. Система запускает их, соответственно, при включении и при выключении доступа для отдельных абонентов. Обычно эти файлы содержат в себе скрипты управления файрволом для разрешения и запрещения прохождения трафика от определенных абонентов. Файл OnConnect запускается при включении доступа абоненту, а файл OnDisconnect — при выключении. При запуске файла OnConnect в него передаются следующие параметры (по порядку):

1. login — логин абонента;
2. ip — ip-адрес абонента;
3. cash — состояние счета абонента;
4. id — уникальный целочисленный идентификатор абонента (уникальность сохраняется в пределах одного запуска системы);
5. dirs — список открываемых направлений (представлен строкой вида «0101100101», где первая цифра соответствует направлению DIR0 а последняя — DIR9).

При запуске файла OnDisconnect в него передаются следующие параметры (по порядку):

1. login — логин абонента;
2. ip — ip-адрес абонента;
3. cash — состояние счета абонента;
4. id — уникальный целочисленный идентификатор абонента (уникальность сохраняется в пределах одного запуска системы);
5. dirs — список открытых направлений (представлен строкой вида «0101100101», где первая цифра соответствует направлению DIR0 а последняя — DIR9).

Файлы OnUserAdd и OnUserDel запускаются при добавлении и удалении абонента. В качестве параметра в них передается логин абонента.

Файл OnChange запускается при внесении изменений в учетную запись абонента. Его можно использовать для дополнительного журналирования изменений. В качестве параметров при запуске в него передаются (по порядку):

1. login — логин абонента;
 2. parameter — название изменяемого параметра;
 3. old value — старое значение параметра;
 4. new value — новое значение параметра.
-

4.21 Описание конфигурационного файла `stg.conf` для плагина `FreeRADIUS rlm_stg.so`

- `local_port` — необязательный параметр. Указывает, с какого локального порта будет происходить обмен данными с Stargazer.
- `server` — IP-адрес сервера Stargazer.
- `port` — порт для обмена данными со Stargazer.
- `password` — пароль, который будет использован для шифрования обмена данными с Stargazer.

Пример секции параметров `rlm_stg.so` для `radiusd.conf`:

```
stg {  
    local_port = 6667  
    server = localhost  
    port = 6666  
    password = 123456  
}
```

Модуль необходимо прописать в секцию `instantiate`, `authorize`. В секции `authenticate` модуль необходимо прописать в подсекциях `AuthType` до соответствующих модулей аутентификации. Например:

```
Auth-Type PAP {  
    stg  
    pap  
}
```

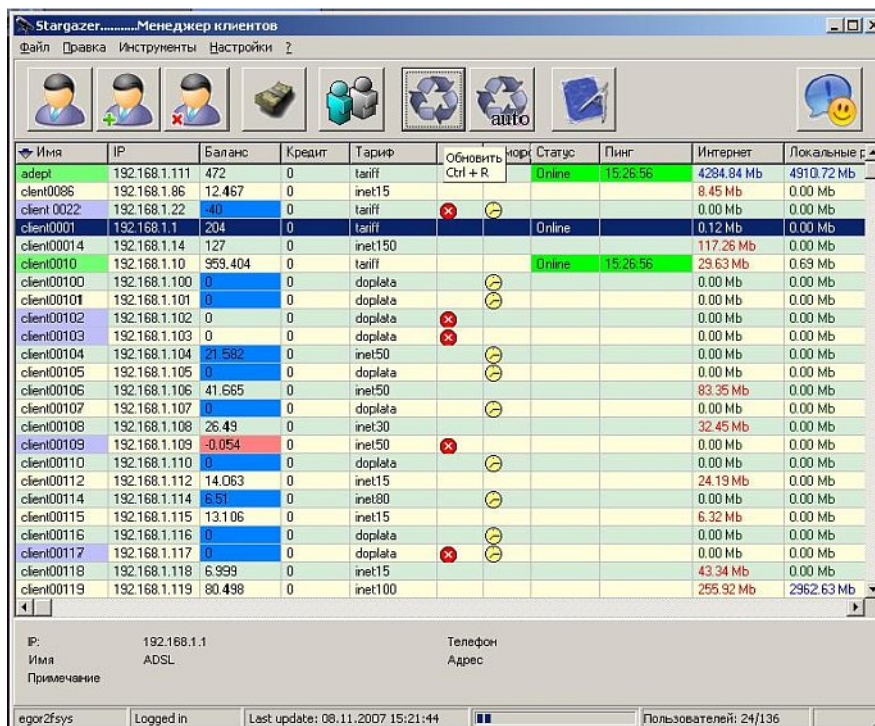
Также модуль должен быть прописан в секции `accounting` и `post-auth`.

Глава 5

Описание работы с конфигуратором

Как и говорилось выше, конфигуратор предназначен для выполнения различных административных действий над клиентами и над некоторыми параметрами сервера.

Общий вид окна конфигуратора при работе показан ниже:



В окне расположены: меню, кнопки быстрого доступа к элементам меню, таблица с информацией о клиентах, панель с дополнительной информацией о клиенте, панель состояния. Следует помнить что почти все команды, вызываемые из конфигуратора будут работоспособны в случае успешного входа на сервер. При этом в панели состояния изменится статус, и появиться имя пользователя, который вошел в систему:



Обзор меню:

- Файл — Сохранить — команда сохраняет таблицу с клиентами в указанный текстовый файл по указанному пути.

- Файл — Login... — команда, предназначенная для доступа на сервер. При этом выведется следующее окно, в котором необходимо ввести учетные данные администратора, как они заданы при вводе администратора:



В случае успешного входа конфигуратор отобразит список клиентов, и даст, в зависимости от ограничений администратора доступ к функциям. После первой установки и старта системы, по-умолчанию логин и пароль на вход равны admin и 123456 соответственно.

- Файл — Logout... — команда, выполняющая выход из системы, при этом таблица клиентов очищается.
- Файл — Печатать всех... — команда выводит предварительный просмотр списка клиентов с указанием трафика по направлениям.
- Файл — Выход — выполняет выход из системы и закрытие окна конфигулятора.
- Правка — Редактировать пользователя... — команда предназначена редактирования данных клиента (дублируется кнопкой



), при её вызове будет вызвано следующее окно:

Редактирование данных пользователя

Логин: egor2fsys

Пароль: ++++++

Еще раз пароль: ++++++

IP адрес(a): 192.168.1.149 ?

Группа:

Телефон:

e-mail: egor2fsys@mail.ru

Имя (реальное):

Адрес:

Примечания:

Деньги: 474.03373 ☐ Добавить ☒ Установить

Запись в лог

Предоплаченный трафик: -342.99199 Деньги

10 Кредит Таймаут кредита

Тарифный план: Svoi

Опциональное поле

Опция 2: https://club/cgi-bin/stat.cgi?sid

UserData2:

UserData3:

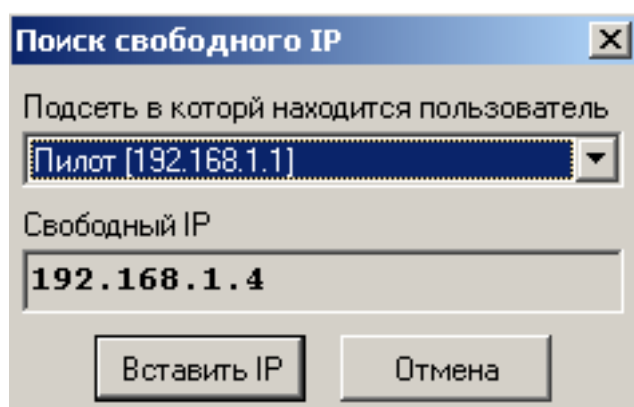
UserData4:

☐ Всегда Online ☐ Отключен ☐ Заморожен

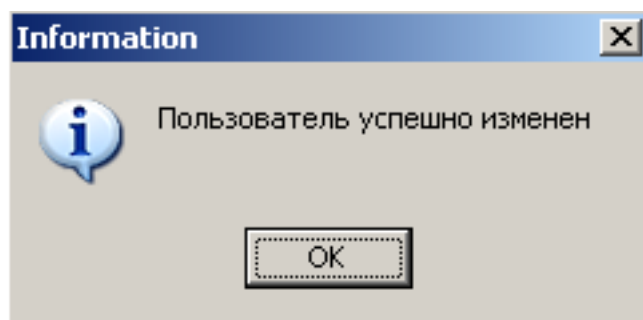
	Отправлено	Принято
Интернет	32988198	479100454
Локаль	1262772834	18477907805
Sky	0	0
Прочий инет	62141777	868965242

В данном окне отображаются все настройки клиента, так как они будут записаны в конфигурационный файл. Опции и настройки данного окна описаны выше. Все числовые величины должны разделяться только знаком «.» (точка), IP адреса разделяются символом «.» (запятая), если клиенту необходимо иметь доступ с любого IP адреса, то в это поле необходимо занести символ * (звездочка). В нижней части окна расположена индивидуальная статистика клиента, её можно изменять, для этого необходимо два раза щелкнуть на подлежащему изменению числе, после редактирование подтвердить изменение кнопкой «Enter». Кнопка «?» предназначена для автоматического поиска свободного IP адреса из указанного в настройках

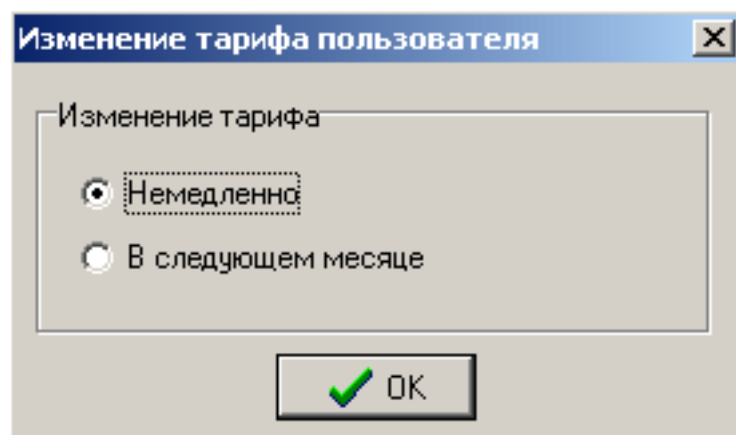
диапазона:



Признак «Заморожен» предназначен для того, чтобы на определенное время прекратить любые денежные действия со счетом клиента, при установке этого признака со счета клиента снимается сумма, указанная в тарифе. Для того чтобы увидеть пароль клиента, сгенерированный кнопкой «Сгенерировать пароль», необходимо нажать на кнопку «Показать пароль», при этом в первом окне пароля отобразится текущий пароль (через 10 секунд он сам исчезнет снова). Следует помнить что при записи пароли и в первом поле ввода и во втором должны совпадать, иначе система выведет предупреждение и не даст записать данные на клиента. Если данные после изменения успешно записаны, то будет выведено сообщение:



В случае изменения тарифа будет выведен вопрос о времени перевода клиента на новый вид тарифа:



Если был выбран вариант «Немедленно», то тариф применится к клиентскому аккаунту немедленно, иначе в случае выбора варианта «В следующем месяце» тариф применится с 1 числа следующего месяца автоматически, при этом строка с клиентом, у которого изменится тариф примет вид:

test	192.168.1.2	5	10	tariff/cccc	группа 1
------	-------------	---	----	-------------	----------

Т. е. в колонке «Тариф» появится запись двойного тарифа, где перед знаком «/» находится текущий, а после – тот тариф, который будет применен со следующего месяца. Следует учи-

тывать что при переходе на другой тариф в течении месяца (выбрана опция «Немедленно») система пересчитает средства на счёте следующим образом: если в тарифе есть абонплата, то с переходом на новый тариф с клиента будет снята часть абонплаты от того тарифа с которого переходят и в конце месяца с него будет снята часть абонплаты от того тарифа на который перевели. Аналогично клиент получит только часть бесплатных Мб от того тарифа на который его перевели.

- Правка - Добавить пользователя... – команда, позволяющая добавить нового клиента в систему (дублируется кнопкой



). При вызове этой команды будет выведено окно для заполнения свойств клиента:

Создание нового пользователя

Логин:

Пароль:

Еще раз пароль:

IP адрес(a): ?

Группа:

Телефон:

e-mail:

Имя (реальное):

Адрес:

Примечания:

Деньги: ☐ Добавить ☒ Установить

Запись в лог

Предоплаченный трафик:

Кредит

Тарифный план:

Опциональное поле

Опция 2

UserData2

UserData3

UserData4

☐ Всегда Online ☐ Отключен ☐ Заморожен

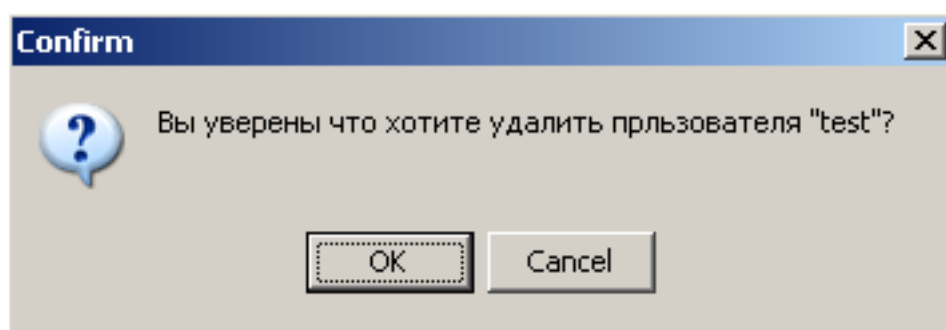
	Отправлено	Принято
Интернет	0	0
Локаль	0	0
Sky	0	0
Прочий инет	0	0

Обязательными к заполнению являются поля: логин, пароль и его подтверждение, IP-адрес(a), тарифный план. Остальные поля и опции являются необязательными. В случае если какой то обязательный параметр не заполнен, система сообщит об ошибке и предложит ввести недостающие данные, если же все данные указаны верно, то система запишет нового клиента в систему. Также при добавлении нового клиента система выставит количество бесплатных Мб которое стоит в тарифе, установленном клиенту, пропорционально времени оставшемуся до обнуления статистики. Также с клиента будет снята абонплата выставленная в тарифе пропорционально времени до конца месяца.

- Правка - Удалить пользователя... – команда позволяющая удалить клиента из системы, при этом все клиентские настройки не удаляются физически на сервере, а переносятся в специальную папку /deleted_users. Команда дублируется кнопкой



. При вызове этой команды будет вызвано подтверждение на удаление:



- Правка - Тарифные планы... – команда вызывающая редактор тарифных планов (дублируется кнопкой



):

Редактор тарифных планов

Абонплата: 15.000
Предоплаченных Мб: 10.000
Подсчет трафика: Upload+Download
Стоимость заморозки: 5.000

Тарифные планы:
Dizainer
Doplata
Inet100
Inet15
Inet200
Inet30
Inet50
Svoi

Сохранить
Добавить
Удалить

Интернет
Часы: 07, Минуты: 45, День: 1.0000, Цена днем: 100, Порог (Мб):
Часы: 00, Минуты: 15, Ночь: 0.6000, Цена ночью: ☒ Без порога
☐ Цена не зависит от времени

Локаль
Часы: 01, Минуты: 32, День: 0.0010, Цена днем: 200, Порог (Мб):
Часы: 10, Минуты: 20, Ночь: 0.0320/0.0340, Цена ночью: ☒ Без порога
☒ Цена не зависит от времени

Бесплатно
Часы: 02, Минуты: 33, День: 0.0520/0.0540, Цена днем: 300, Порог (Мб):
Часы: 11, Минуты: 30, Ночь: , Цена ночью: ☐ Без порога
☒ Цена не зависит от времени

Прочий инет
Часы: 03, Минуты: 30, День: 0.0720/0.0740, Цена днем: 50, Порог (Мб):
Часы: 12, Минуты: 40, Ночь: 0.0010/0.0005, Цена ночью: ☐ Без порога
☐ Цена не зависит от времени

Ok

Цены вводятся через знак «/» - это разделение цены до/после перехода порога по направлению, если цена будет введена без знака «/», то она будет применена как к трафику до, так и после превышения одинаково. Порог устанавливается в поле «Порог (Мб)», если установлен признак «Без порога», то цена вводится одна. В стоимость заморозки устанавливается сумма, которая будет снята с клиента при установке признака «Заморожен».

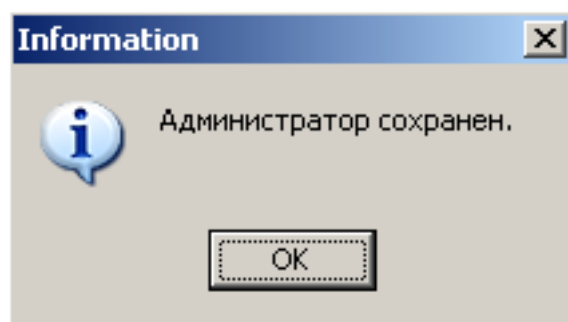
После изменения тарифного плана необходимо нажать кнопку «Сохранить» для записи данных. Измененный тарифный план начинает применяться сразу после записи.

- Правка - Администраторы - команда, позволяющая изменять права администраторов и добавлять/удалять их. Дублируется кнопкой



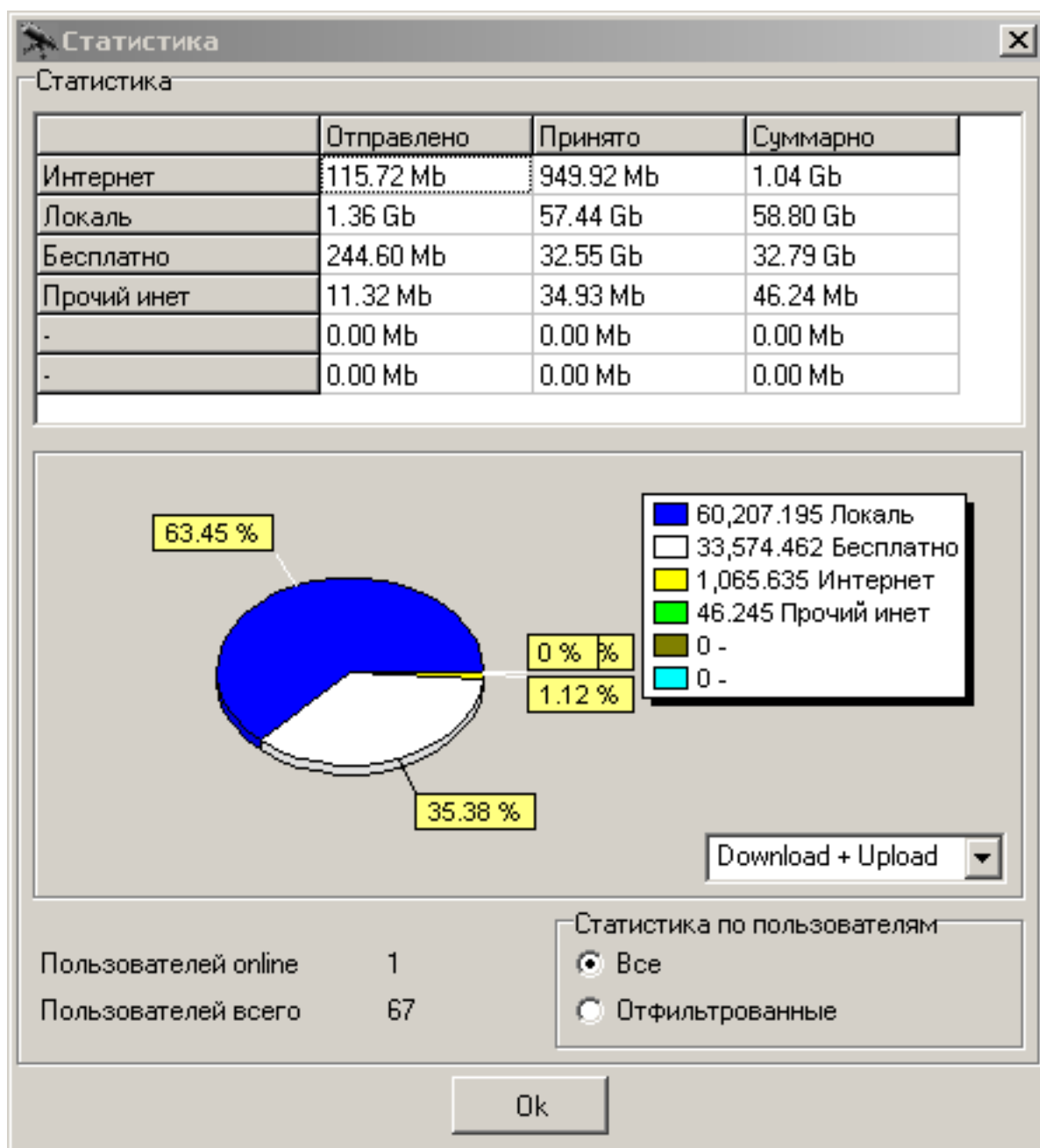
, при вызове команды будет выдано окно со списком текущих администраторов:

Пароль и подтверждение пароля в целях безопасности не отображаются, при записи если оставить незаполненными поля пароля, то он изменен не будет. Для сохранения сделанных изменений необходимо нажать кнопку «Сохранить». В случае успешного сохранения данных администратора будет выведено сообщение:



Если имела место какая-либо ошибка, сервер сообщит об ней.

- Инструменты - Статистика - команда отображает общую статистику по направлениям, с визуальным обзором на диаграмме:



При выборе вариантов внизу соответственно будет перерисована диаграмма. Также можно указать по каким клиентам показывать суммарную статистику: по всем или только по отфильтрованным, и какую собственно на диаграмме прорисовывать статистику.

- Инструменты - Фильтр - команда позволяет отфильтровать текущую таблицу с данными о клиентах по заданному фильтру:

Фильтр

Логин
[Text Field]

Тариф
[Dropdown Menu]

Группа
[Dropdown Menu]

Деньги
Больше [Text Field]
Меньше [Text Field]

☐ Online ☐ Отключен ☐ Не отключен
☐ Не Online ☐ Заморожен ☐ Не заморожен

Имя [Text Field] Адрес [Text Field]

Телефон [Text Field] E-mail [Text Field]

Примечания [Text Field] IP [Text Field]

Очистить

Условия фильтрации
☒ И ☐ Или

Ok Cancel

? - один любой символ
* - любое кол-во любых символов

- Инструменты - Разослать письма... – команда позволяет разослать письма указанным клиентам с сервера:

Расылка писем

Загрузить шаблон Сохранить шаблон Подстановки Отправить Просмотр

☒ absinth
☒ added
☒ boris
☒ qwerty
☒ sj
☒ test
☐ test1
☐ test2
☒ troyanda
☒ web_arsenal

Все Никто Должники Инверсия

From: admin@my.net
Subject: Состояние счета

Уважаемый пользователь %name!

По состоянию на %date %time на вашем счету осталось %cash y.e.

Перекачано по миру %udload0
Перекачано по городу %udload1

По всем вопросам обращайтесь по телефону 123-45-67
С уважением, администрация сети my.net

Заккрыть

В поля From, Subject заносятся электронный адрес отправителя и тема писем соответственно. Кнопка «Все» позволяет выделить всех клиентов из списка, «Никто» - убирает выделение со всех в списке, «Должники» - выделяет клиентов, у которых баланс меньше нуля и превысил «Кредит», «Инверсия» - позволяет инвертировать выделение, т. е. все кто был помеченными станут непомеченными и наоборот. Кнопки «Загрузить шаблон» и «Сохранить шаблон» позволяют загружать исходный и сохранять текущий шаблоны соответственно. В «Подстановках» можно выбрать спец. Макросы, которые потом при отправке каждому клиенту заменяться на реальные для него цифры или информацию. Пример сохраненного шаблона:

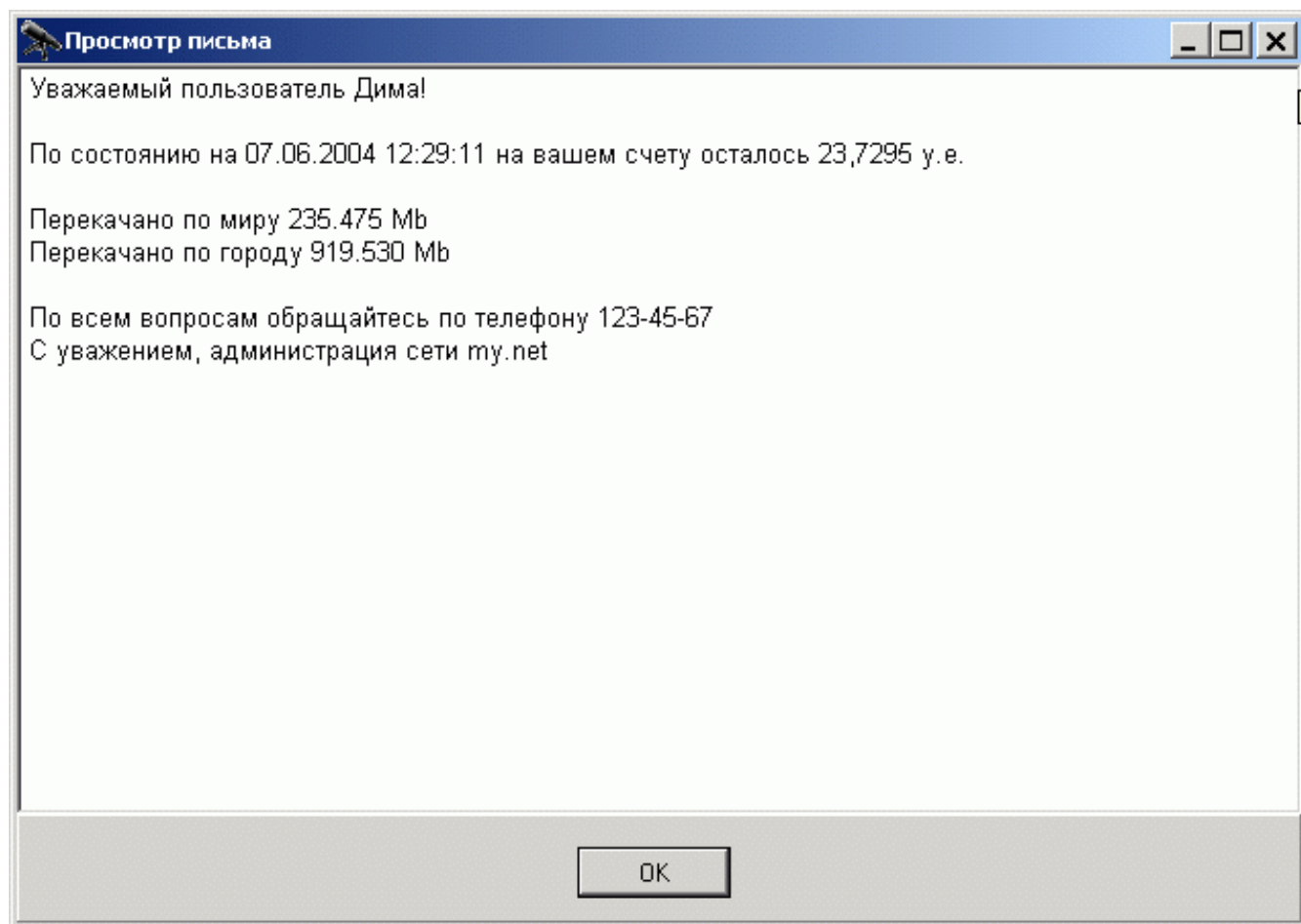
```
admin@my.net
Состояние счета
Уважаемый пользователь %name!

По состоянию на %date %time на вашем счету осталось %cash y.e.

Перекачано по миру %udload0
Перекачано по городу %udload1

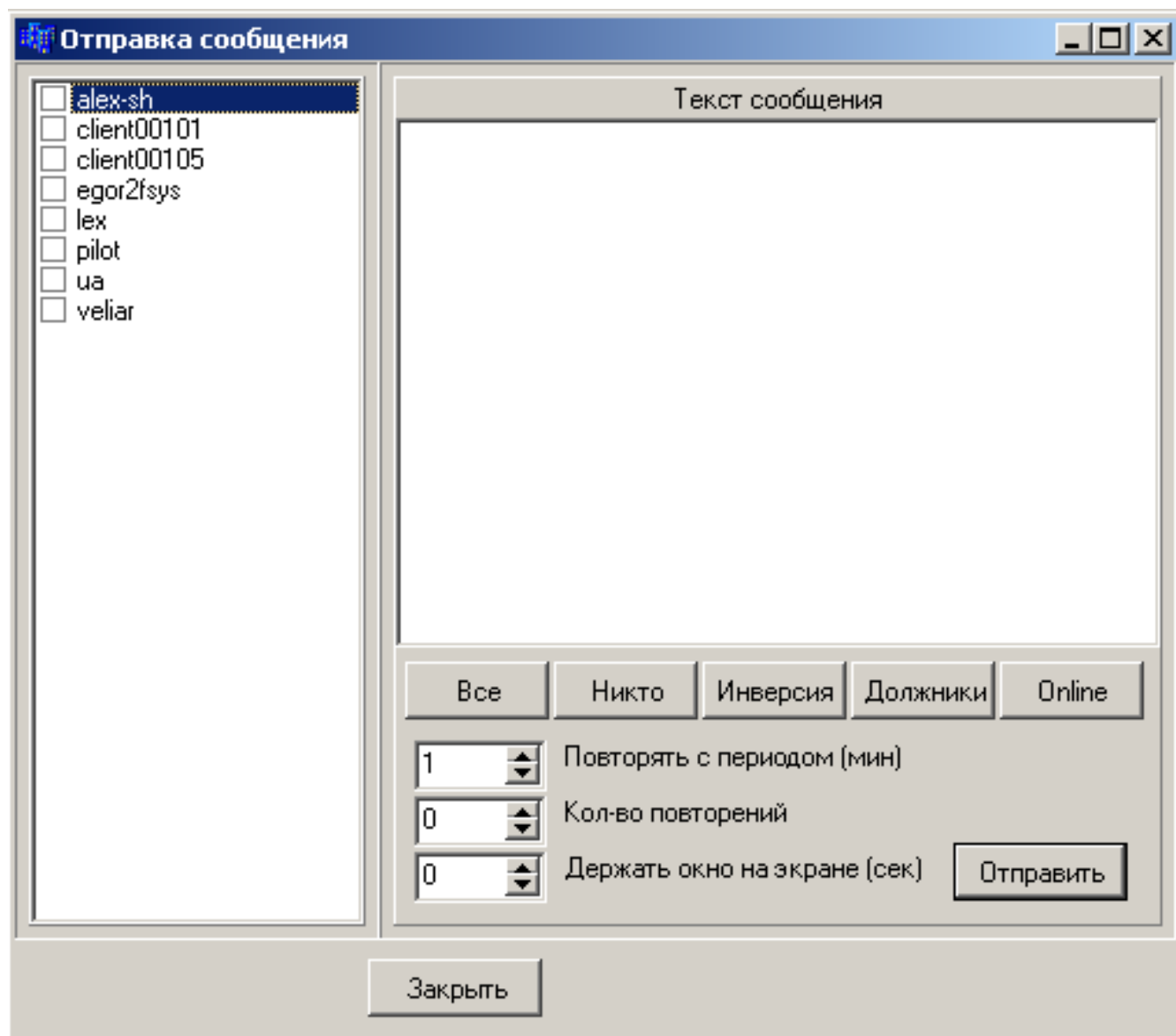
С уважением, администрация сети my.net
```

Кнопка «Просмотр» позволяет просмотреть итоговое сообщение для выбранного клиента:



Следует помнить что для корректной рассылки писем в настройках необходимо указать SMTP сервер. Рассылка писем предусматривает сохранение и восстановление шаблонов.

- Инструменты - Отправить сообщение - команда предназначена для отправки сообщений клиентам, в данный момент находящимся в OnLine или в OffLine (следует учитывать, что сообщение дойдет моментально только лишь в том случае, если клиент пользуется в данный момент авторизатором, если же клиент в данный момент находится в состоянии OffLine, то сообщение дойдет сразу же после первого подключения):



При отправке сообщения можно задать пользователей, которым уйдет сообщение, интервал повторения сообщения в минутах, количество повторений, а также время в секундах, в течение которого окно с сообщением будет держаться на экране у пользователя.

- Инструменты – Очистить таблицу пользователей – команда позволяет очистить таблицу клиентов и их статистики, однако следует понимать что таблица очищается только в окне конфигуратора, никакие данные на сервере изменены не будут.
- Настройки – Настройки... – команда вызывает окно с настройками конфигуратора (дублируется кнопкой



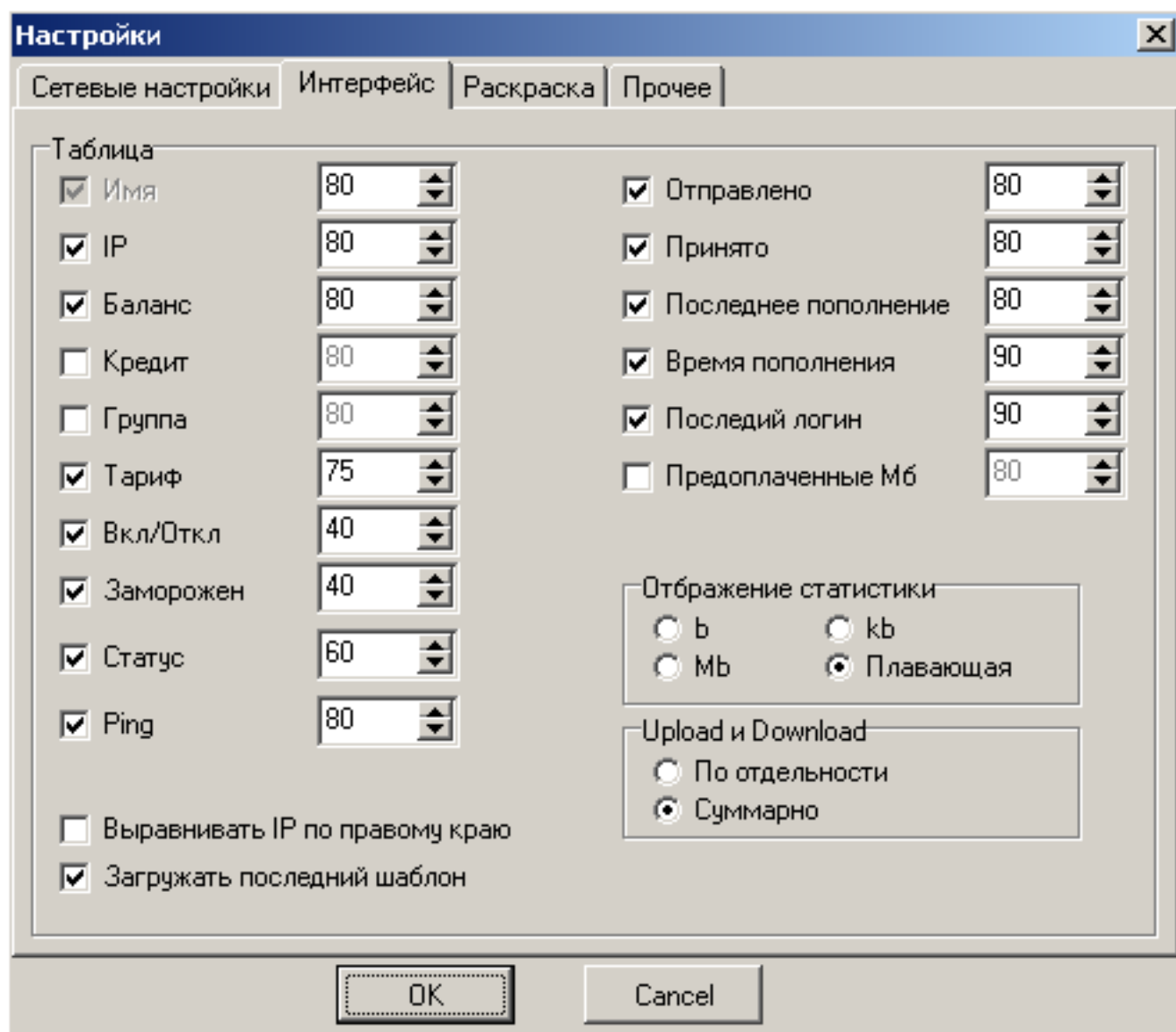
):

The screenshot shows a Windows-style dialog box titled "Настройки" (Settings). It has four tabs: "Сетевые настройки" (Network settings), "Интерфейс" (Interface), "Раскраска" (Skin), and "Прочее" (Miscellaneous). The "Интерфейс" tab is selected. The dialog is divided into several sections:

- Сервер** (Server):
 - Имя сервера (Server name): 192.168.1.18
 - Порт (Port): 4444
- Ping**:
 - Коррекция времени (сек) (Time correction (sec)): 0
- Автообновление** (Auto-update):
 - 5 сек (5 sec)
- SMTP**:
 - SMTP сервер (SMTP server): smtp.mail.ru
 - SMTP порт (SMTP port): 25
- Длина генерируемого пароля** (Generated password length): 8

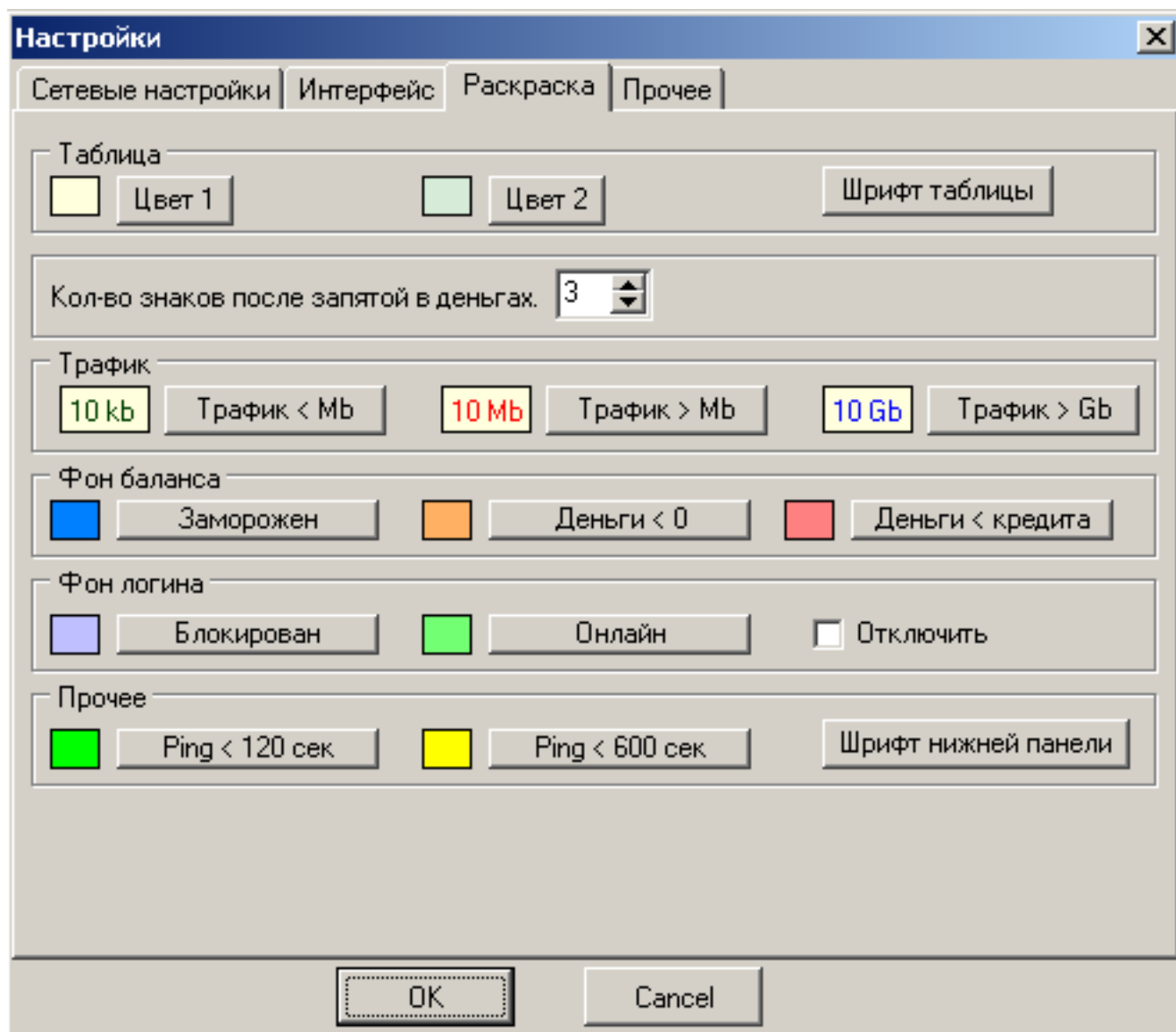
At the bottom are "OK" and "Cancel" buttons.

В данном окне задаются настройки конфигулятора: адрес сервера, порт, адрес почтового сервера и его порт, интервал авто обновления, длина генерируемого пароля и прочие параметры конфигулятора. На закладке «Интерфейс»

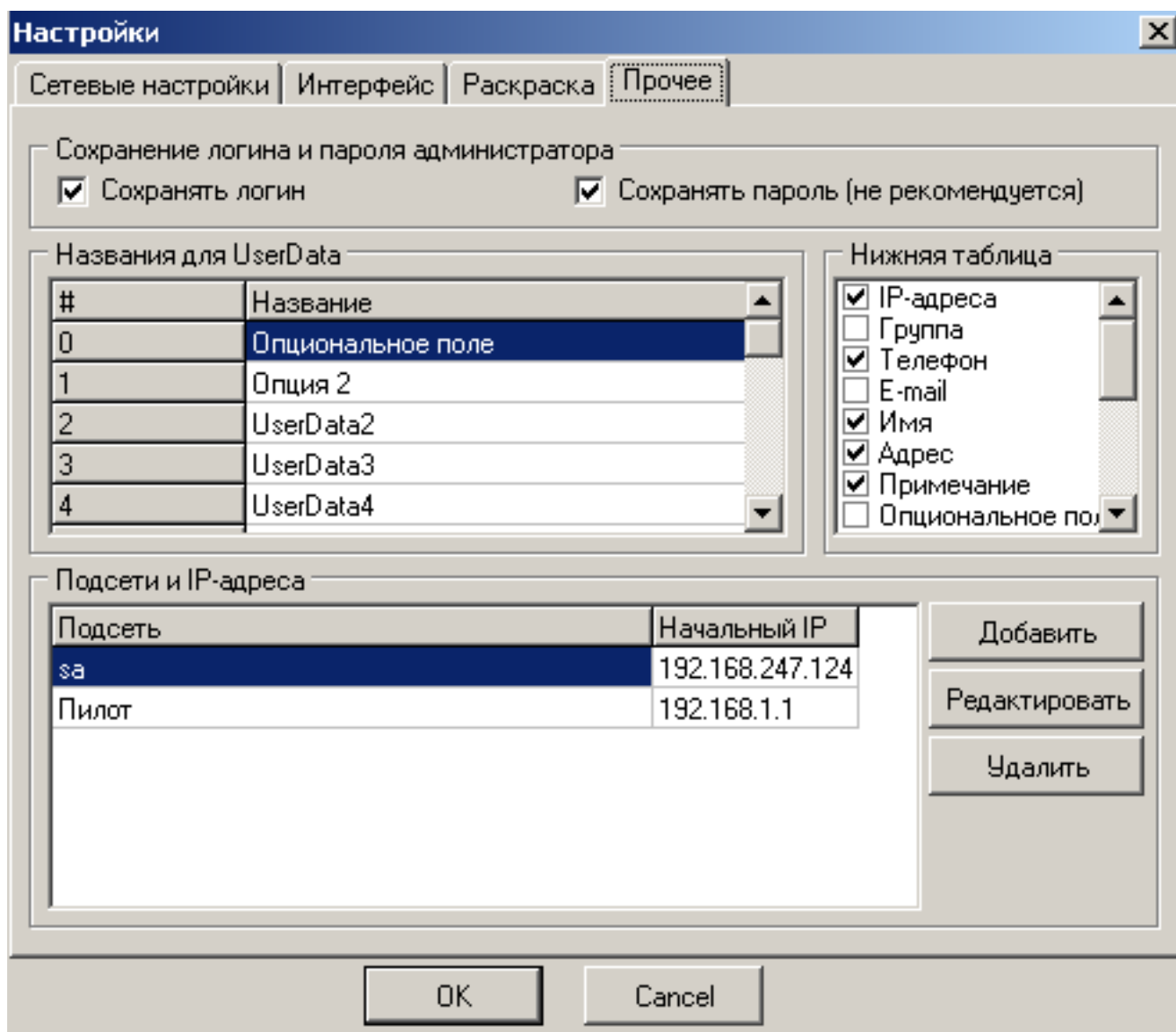


задаются: видимость колонок в окне клиентов, а так же ширина. Признак «Загружать последний шаблон» определяет загружать ли конфигуратору последний шаблон, который был использован при последней отправке писем клиентам.

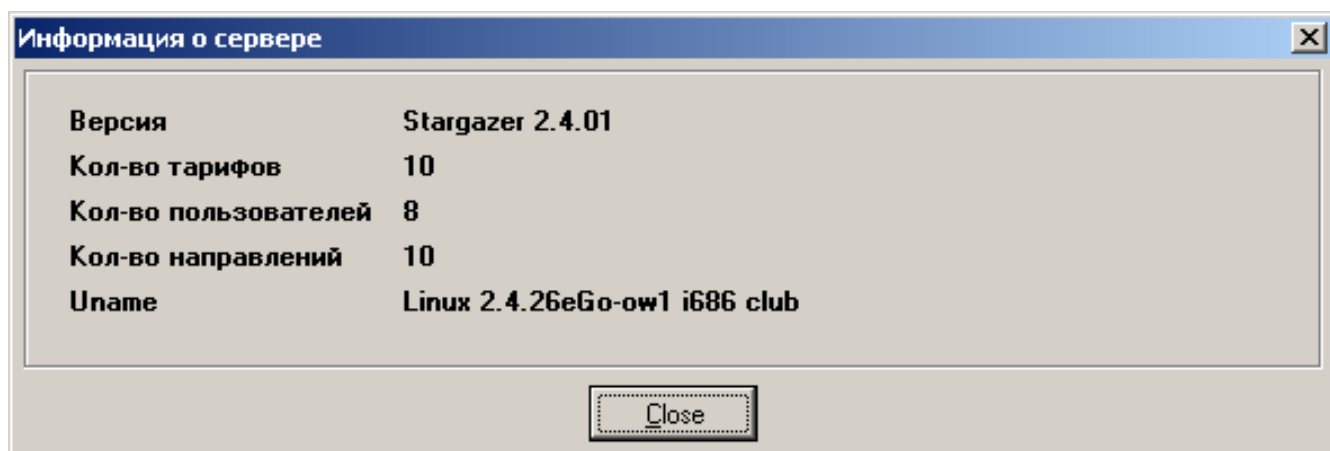
На закладке «Раскраска» можно задать цвета, шрифт и другие параметры отображения в основной таблице и в панели информации.



На закладке «Прочее» можно указать, сохранять ли логин и (или) пароль администратора. Пароль шифруется и привязывается к HDD компьютера, однако, НЕ РЕКОМЕНДУЕТСЯ сохранять пароль администратора !!! Также здесь можно задать имена полей типа UserDataN для отображения в конфигураторе и параметры, которые будут показываться в нижней панели информации. В разделе «Подсети и IP-адреса» можно задать имя и начальный адрес для автоматического поиска свободного IP адреса.



- Настройки - Информация о сервере - команда позволяет узнать некоторую информацию о сервере, о его версии и некоторые другие данные:



Кнопка



позволяет обновить данные о клиентах в таблице.

Кнопка



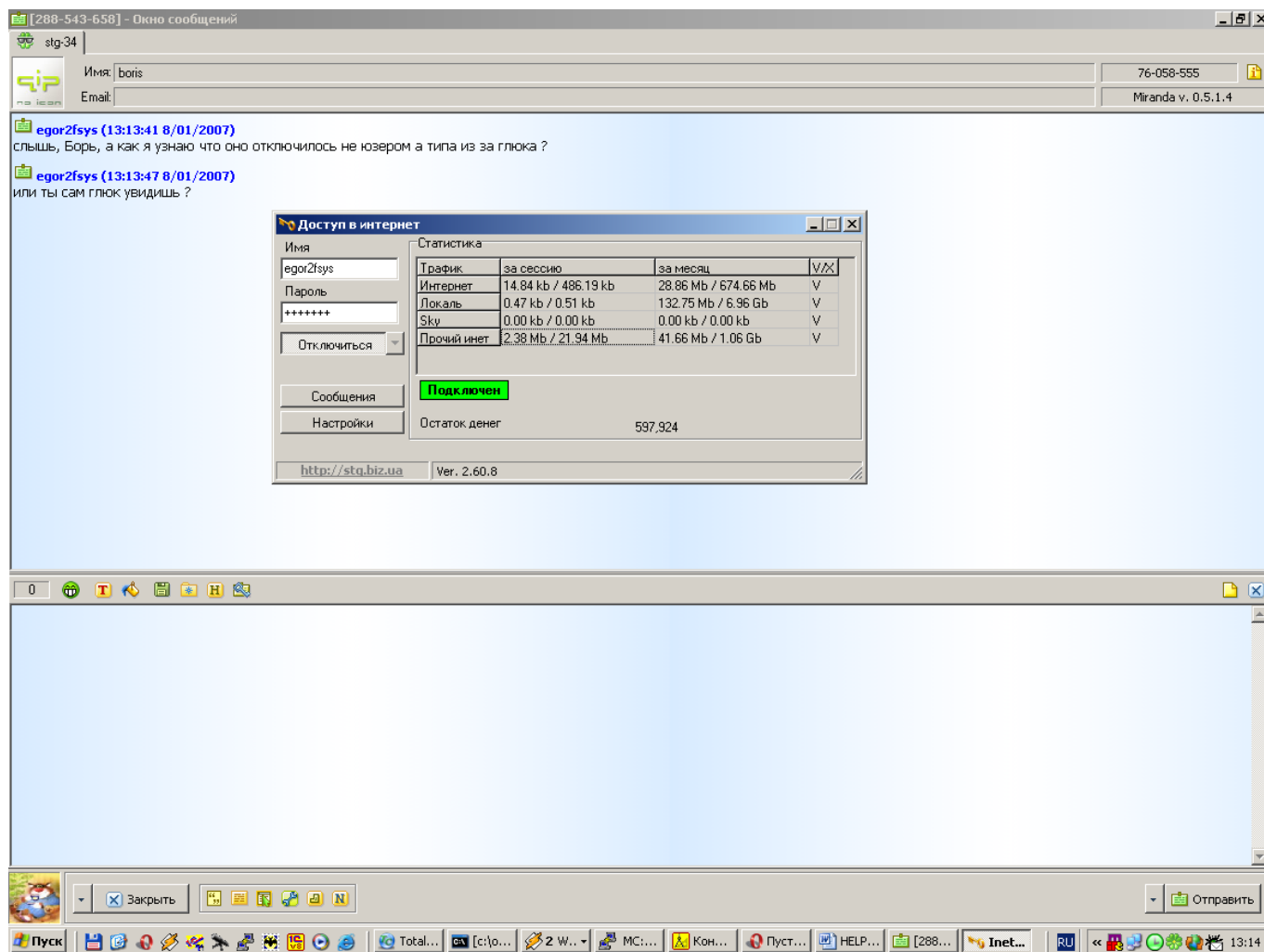
позволяет включать или отключать авто обновление данных. В настройках конфигулятора можно задать интервал авто обновления.

Глава 6

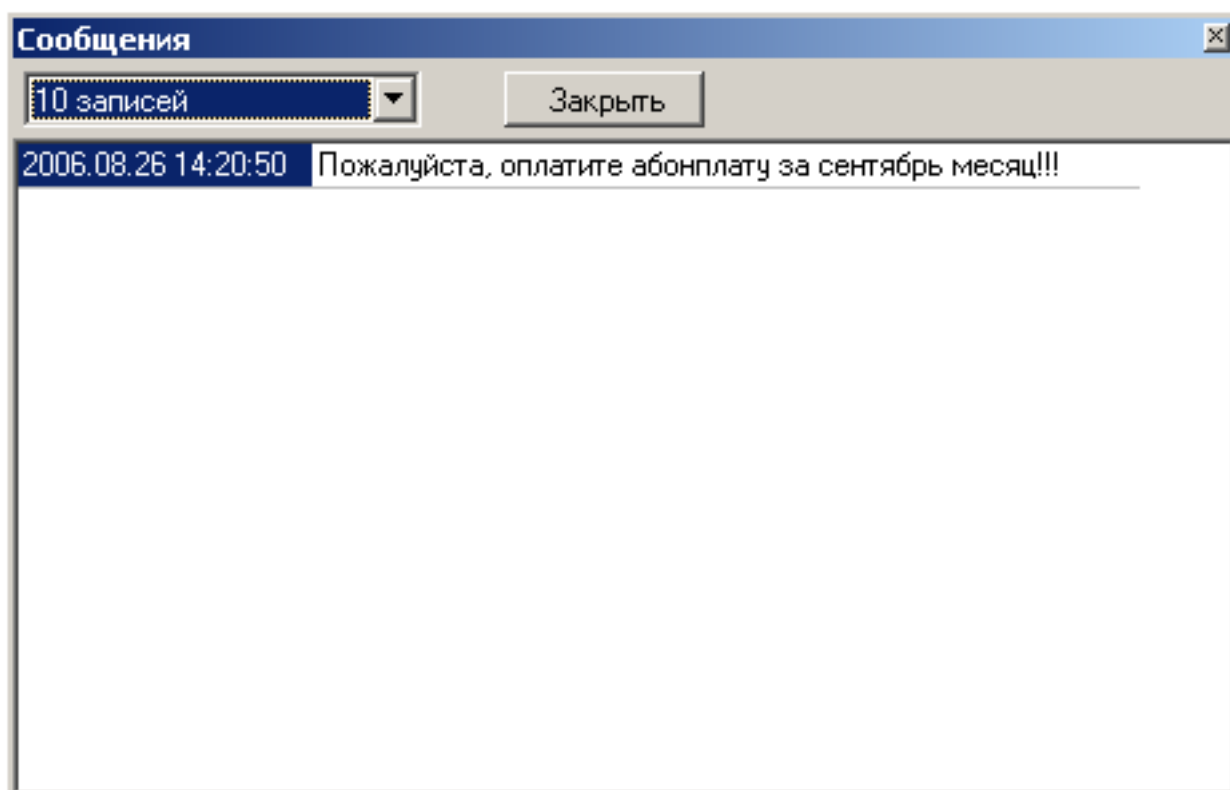
Описание работы с авторизатором InetAccess.

Клиентский авторизатор предназначен для выдачи клиентам сети. С помощью него возможно узнать состояние счета, а так же состояние перекачанного трафика по направлениям. Сервер системы может работать со старыми версиями авторизаторов, т. е. сохраняется обратная совместимость, однако по мере возможности необходимо обновлять клиентские авторизаторы до последних версий.

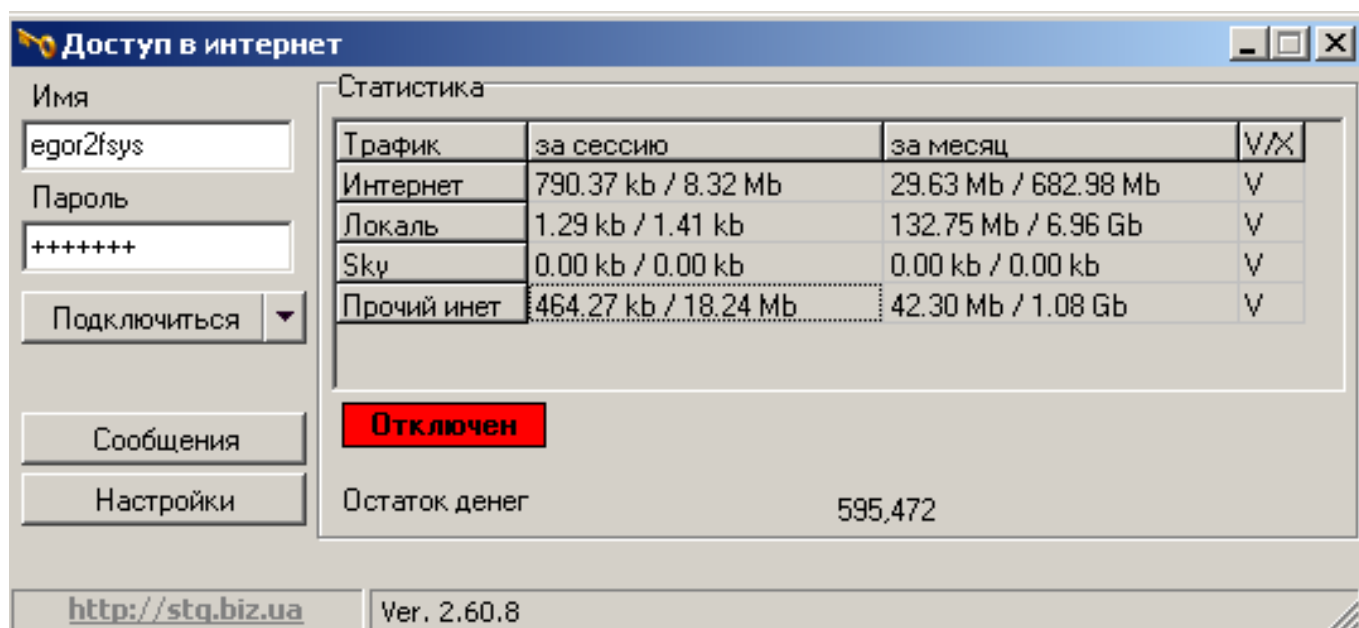
Основное окно авторизатора показано ниже:



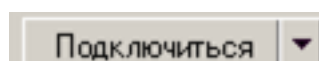
В поля Имя и Пароль вводятся, выданные администрацией сети, логин и пароль соответственно. Кнопка «Сообщения» служит для просмотра истории сообщений, отправленных администрацией сети:



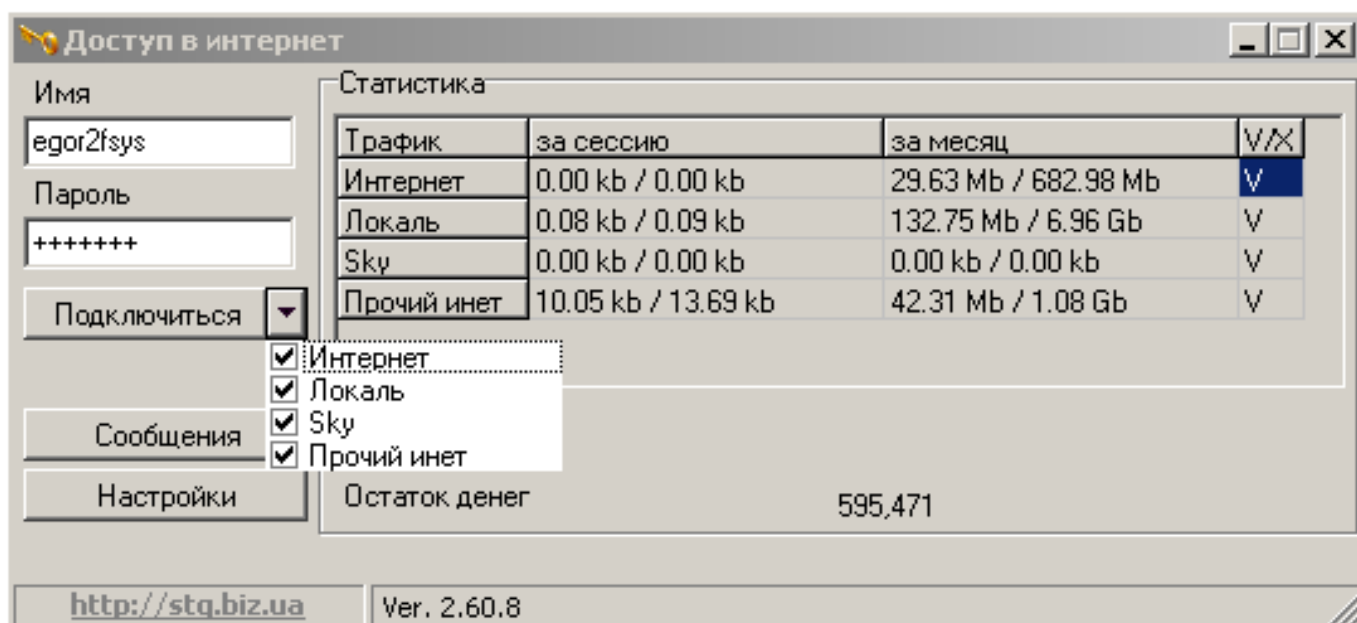
В отключенном состоянии авторизатор имеет следующий вид:



Для того чтобы авторизоваться, необходимо заполнить все поля, выбрать при необходимости те направления, к которым следует подключиться и нажать кнопку «Подключиться». Направления необходимо выбрать с помощью кнопки

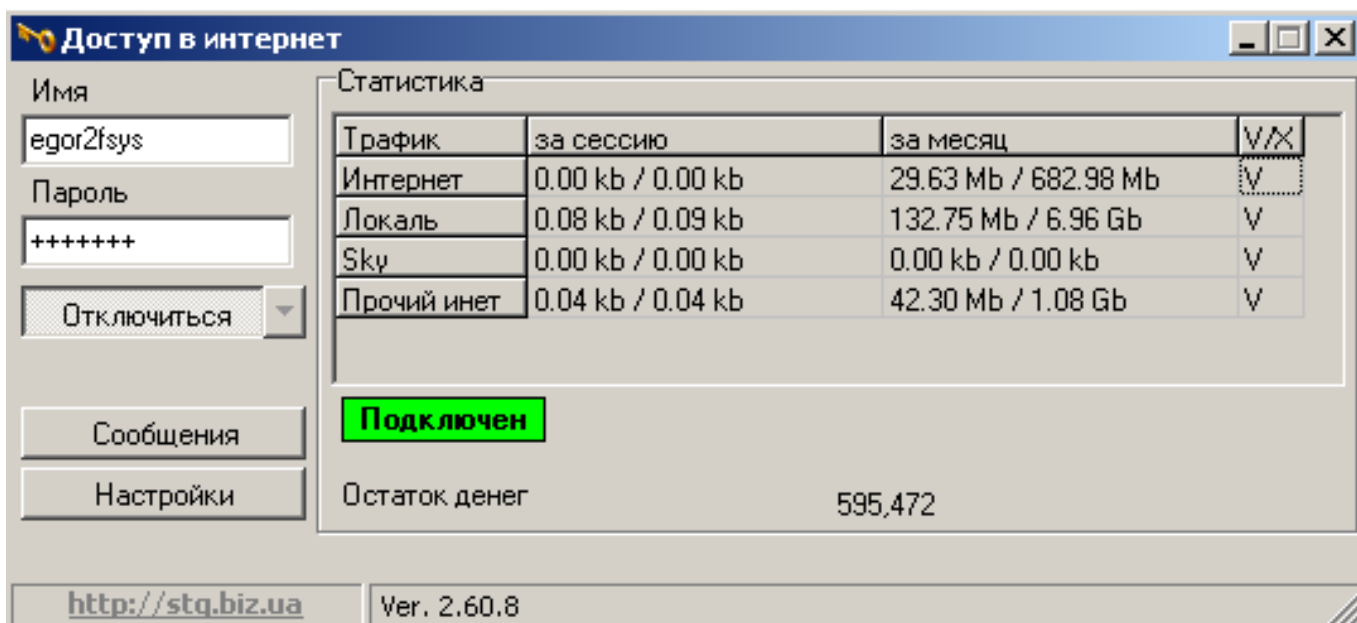


до подключения:

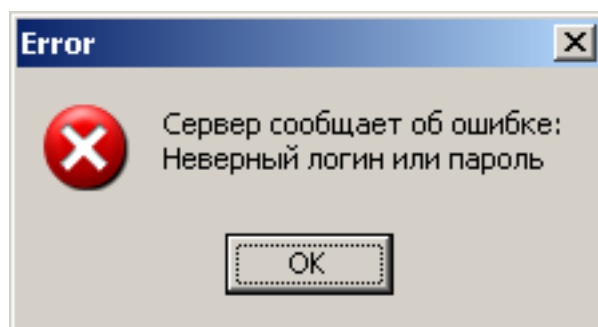


После подключения направления, выбранные для подключения будут отмечены символом «V», а направления к которым подключаться не требуется, будут отмечены символом «X» в колонке авторизатора «V/X».

При успешном подключении окно авторизатора примет следующий вид:



Если же по каким-то причинам подключение не удастся, то авторизатор сообщит об этом:



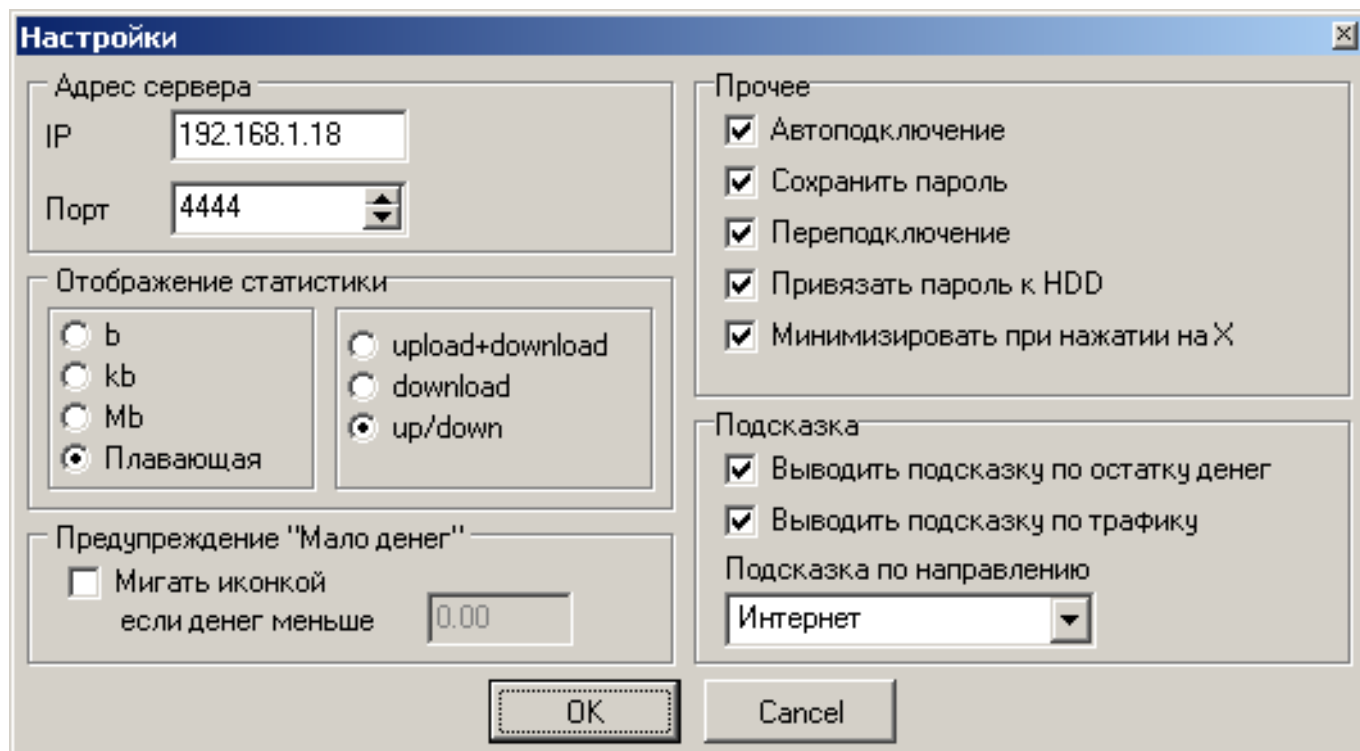
При этом будет сообщено именно о той ошибке, которая имеет место в вашей ситуации.

6.1 Описание настроек авторизатора.

Для вызова диалога настроек авторизатора нажмите кнопку

Настройки

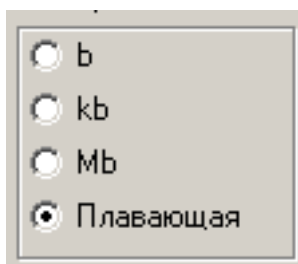
в окне. Откроется следующий диалог:



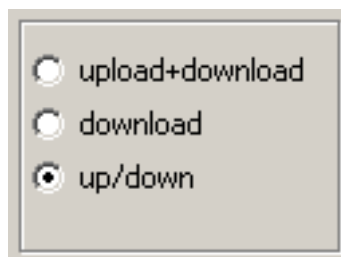
В данном окне сосредоточены все настройки авторизатора:

- IP – адрес сервера, к которому производится подключение. Выдается администрацией сети.
- Порт – порт сервера, на который будет обращаться программа авторизации, чаще всего 5555. Выдается администрацией сети.
- Признак «Автоподключение» позволяет автоматически выполнять подключение к серверу при старте авторизатора.
- Признак «Сохранять пароль» позволяет сохранять пароль клиента в файле настройки авторизатора. Следует помнить, что если признак «Сохранять пароль» не установлен, то признак «Автоподключение» автоматически становится неактивным.
- Признак «Переподключение» позволяет автоматически переподключаться в случае обрыва связи с сервером или в других непредвиденных случаях.
- Признак «Привязывать пароль к HDD» позволяет зашифровать сохраненный в файле настройки пароль. При этом даже произойдет кража файла с паролем, то он не сможет быть декодирован на компьютере с другим HDD. Очень рекомендуется выставлять опцию включенной.
- Признак «Минимизировать при нажатии на X» позволяет при закрытии главного окна авторизатора, не выгружать программу из памяти, а сворачивать её в системный трей.

- Отображение статистики:



- данное меню позволяет выбрать, в каком виде измерений будут выводиться данные о прошедшем трафике. Если выбрано «Плавающая», то данные будут отражаться в зависимости от объёма в тех или иных единицах.



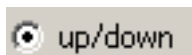
- данное меню позволяет выбирать каким образом будут отображаться данные о прошедшем трафике в разрезе «отправлено» и «принято». Если выбрана опция



, то отображаться будет сумма обоих трафиков, как отправленного, так и принятого. Если выбрана опция



то отражаться будет только принятая часть трафика. И если выбрана опция



, то отражаться трафик будет отдельно.

- Признаки «Выводить подсказку по остатку денег» и «Выводить подсказку по трафику» - позволяют выводить при наведении на значок в системном трее информацию об остатке денежных средств на счету и об переданном трафике соответственно. Для опции «Выводить подсказку по трафику» необходимо выбрать направление, по которому будет выводиться подсказка.
- Признак «Мигать иконкой, если денег меньше» отвечает за включение сигнализации, если сумма денег на балансе клиента станет меньше установленной.

В конфигурационном файле авторизатора (InetAccess.ini) можно указать параметры, которые не могут быть изменены с помощью окна «Настройка». Параметр IPEditable, если он принимает значение 1, то в окне настроек авторизатора становится недоступным поле IP адрес сервера. Параметр PortEditable, если он принимает значение 1, то в окне настроек авторизатора становится недоступным поле порт сервера. Рекомендуется при выдаче пользователям авторизатора, выдавать и файл настроек с прописанными значениями IP и порта сервера биллинга, а также прописывать IPEditable=1 и PortEditable=1. Параметр NeverSavePassword отвечает за возможность установки флага «Сохранять пароль». Если значение данного параметра равно 1, то авторизатор не позволит установить этот признак, и соответственно не будет сохранять и восстанавливать пароль. Параметр CURRENCY позволяет задать название валюты для отображения в авторизаторе, в которой ведется учет в биллинговой системе.

Авторизатор имеет возможность смены иконок, которые будут отображаться в системном трее. Для того, чтобы авторизатор принял пользовательские иконки, следует в том каталоге, где расположен InetAccess.exe создать папку iso и положить туда файлы с иконками. Файлы должны

иметь определенные имена: `ia_green.bmp`, `ia_grey.bmp`, `ia_red.bmp`, `ia_yellow.bmp`. Если пользовательские иконки не будут заданы или будет отсутствовать папка `ico`, то будут использованы стандартные иконки, встроенные в авторизатор.

Глава 7

Описание работы с консольным авторизатором

Надо-бы написать

Глава 8

Описание работы с консольным конфигуратором

Консольный конфигуратор позволяет добавлять, удалять пользователей, получать и устанавливать некоторые параметры пользователя и отправлять ему текстовые сообщения. Общий формат вызова для получения значения параметра:

```
sgconf get -s <server> -p <port> -a <admin> -w <admin_password> -u <user> <options>
```

- `server` — адрес сервера;
- `port` — порт конфигуратора на сервере;
- `admin` — логин администратора на сервере;
- `admin_password` — пароль администратора на сервере.
- `user` — логин пользователя на сервере.

Общий формат вызова для установки параметра, добавления или удаления пользователя:

```
sgconf get -s <server> -p <port> -a <admin> -w <admin_password> -u <user> <options>
```

Параметры вызова — такие-же, что и при получении значения параметра.

- `-c <add_cash[:log_message]>` — добавление денег на счет пользователя. `add_cash` — добавляемая сумма, `log_message` — опциональный комментарий для записи в лог.
 - `-v <set_cash[:log_message]>` — установка денег на счету пользователя. `set_cash` — устанавливаемая сумма, `log_message` — опциональный комментарий для записи в лог.
 - `-c` — при использовании без параметра возвращает состояние счета пользователя.
 - `-t <tariff:now|delayed>` — смена тарифа пользователя. `tariff` — название тарифа, `now` — изменить немедленно, `delayed` — изменить в конце месяца.
 - `-t` — при использовании без параметра возвращает текущий тариф пользователя.
 - `-r <credit>` — предоставление пользователю кредита в размере `credit`.
 - `-r` — при использовании без параметров возвращает текущий кредит.
 - `-E <YYYY-MM-DD>` — указание срока истечения кредита пользователю.
-

- -E — при использовании без параметров возвращает текущий срок истечения кредита пользователю.
 - -o <new_password> — установка пользователю пароля new_password.
 - -o — при использовании без параметра возвращает текущий пароль пользователя.
 - -e <prepaid> — установка количества prepaid-трафика.
 - -e — при использовании без параметра возвращает количество оставшегося prepaid-трафика.
 - -I <*ip_addr[,ip_addr]> — установка IP-адреса пользователя (одного, нескольких или произвольного).
 - -I — при использовании без параметра возвращает текущий IP-адрес пользователя.
 - -A <name> — установка реального имени пользователя.
 - -A — при использовании без параметра возвращает реальное имя пользователя.
 - -N <note> — установка комментария к пользователю.
 - -N — при использовании без параметров возвращает комментарий к пользователю.
 - -D <address> — установка реального адреса пользователя.
 - -D — при использовании без параметра возвращает адрес пользователя.
 - -L <email> — установка email-адреса пользователя.
 - -L — при использовании без параметра возвращает адрес Email пользователя.
 - -P <phone> — установка телефона пользователя.
 - -P — при использовании без параметров возвращает телефон пользователя.
 - -G <group> — установка группы пользователя.
 - -G — при использовании без параметра возвращает текущую группу пользователя.
 - -d <1|0> — включение/выключение пользователя.
 - -d — при использовании без параметров возвращает текущее состояние пользователя (включен/выключен, а не online/offline).
 - -i <1|0> — «заморозка»/«разморозка» пользователя.
 - -i — при использовании без параметра возвращает текущее состояние пользователя («заморожен»/«разморожен»).
 - --disable-stat <1|0> — включение/выключение ведения детальной статистики для пользователя.
 - --disable-stat — при использовании без параметра возвращает состояние флага ведения детальной статистики для пользователя.
 - --always-online <1|0> — включение/выключение режима Always Online для пользователя.
 - --always-online — при использовании без параметра возвращает режим пользователя.
 - --u0 <traff> [--u1 <traff> ...] — установка количества исходящего трафика за месяц по направлениям (0 — 9).
 - --u0 [--u1 ...] — при использовании без параметров возвращает количество исходящего трафика по направлениям.
-

- `--d0 <traff> [--d1 <traff> ...]` — установка количества входящего трафика за месяц по направлениям (0 — 9).
 - `--d0 [--d1 ...]` — при использовании без параметров возвращает количество входящего трафика по направлениям.
 - `--ud0 <userdata> [--ud1 <userdata> ...]` — установка полей UserData<0-9>.
 - `--ud0 [--ud1 ...]` — при использовании без параметра возвращает поля UserData<0-9>.
 - `-m <message>` — отправляет пользователю сообщение.
 - `-n` — добавляет нового «пустого» пользователя.
 - `-l` — удаляет пользователя.
-

Глава 9

Лицензия, авторы, копирайты, благодарности.

Система распространяется по лицензии GPL v2.

Авторами системы являются Борис Михайленко aka [stg-34](#), Максим Мороз, Максим Мамонтов aka [madf](#), Игорь Лысейко и другие.

Авторами файлов справки являются Георгий Филонов aka [egor2fsys](#), Максим Мамонтов aka [madf](#) и Елена Шевелева.

В системе использована библиотека [LibExpat](#).

Модуль БД на основе СУБД Firebird использует библиотеки [libfbclient](#) и [IBPP](#). Модуль БД на основе СУБД MySQL использует библиотеку [libmysqlclient_r](#). Модуль БД на основе СУБД PostgreSQL использует библиотеку [libpq](#).

Авторы выражают благодарность всем администраторам и пользователям Stargazer за сообщения об ошибках и предложения новых возможностей системы. Также выражаем благодарность форумам «Локальные сети Украины» и «Домашние сети России».

Глава 10

Приложение А: конвертация БД

Для преобразования базы между любыми двумя модулями хранилища используется утилита `convertor` (исходный код утилиты находится в каталоге `projects/convertor` стандартной поставки системы). Утилита работает из каталога сборки без установки в ОС. Для сборки утилиты необходимо запустить скрипт `./build`.

При запуске утилиты в качестве параметра можно передать путь к конфигурационному файлу. По умолчанию будет использован файл `convertor.conf` из каталога сборки. Конфигурационный файл имеет такой-же формат, что и конфигурационный файл `stargazer`. Глобальный параметр `ModulesPath` указывает местонахождение модулей хранилища. Далее следуют две секции: `<SourceStoreModule>` и `<DestStoreModule>`. Они описывают, соответственно, источник данных и их приемник. Так, например, для переноса всей информации из БД на основе файлов в БД Firebird необходимо в первой секции прописать параметры файлового хранилища, а во второй — хранилища Firebird.

Пример файла конфигурации:

```
# Путь к директории, в которой находятся модули сервера
ModulesPath = ../stargazer/modules

#####
# Store module
# Настройки плагина работающего с БД сервера

# Второй параметр - это имя модуля без mod_ в начале и .so в конце
# Т.е. полное имя модуля mod_store_files.so
<SourceStoreModule store_files>

    # Рабочая директория сервера, тут содержатся данные о тарифах, пользователях,
    # администраторах и т.д.
    WorkDir = /var/stg-test

    # Владелец, группа и права доступа на файлы статистики (stat) пользователя
    ConfOwner = root
    ConfGroup = wheel
    ConfMode = 600

    # Владелец, группа и права доступа на файлы конфигурации (conf) пользователя
    StatOwner = root
    StatGroup = wheel
    StatMode = 640

    # Владелец, группа и права доступа на лог-файлы (log) пользователя
```

```
UserLogOwner = root
UserLogGroup = wheel
UserLogMode = 640

</SourceStoreModule>

<DestStoreModule store_firebird>
  server = localhost
  database = /var/stg/stargazer.fdb
  user = stg
  password = 123456
</DestStoreModule>

#<DestStoreModule store_mysql>
#   # Имя пользователя БД
#   dbuser = stg
#
#   # Пароль пользователя БД
#   rootdbpass = 123456
#
#   # Имя БД на сервере
#   dbname = stg
#
#   # Адрес сервера БД
#   dbhost = localhost
#</DestStoreModule>
```

Глава 11

Приложение Б

11.1 Описание формата файлов `/var/stargazer/admins/*.adm`.

В каталоге `/var/stargazer/admins` (путь может быть изменен в настройках плагина `store_files`) находятся файлы содержащие параметры администраторов системы. Имя файла имеет вид: `<login>.adm` где `<login>` — логин администратора. Файл имеет формат «ключ-значение». Каждая строка файла содержит название параметра и его значение, разделенные знаком «=», или комментарий начинающийся с символа «#». Допустимы следующие параметры:

- `Password` — пароль администратора в зашифрованном виде;
- `ChgStat` — право на изменение статистики абонента, может принимать значения 0 и 1;
- `ChgConf` — право на изменение данных учетной записи абонента, может принимать значения 0 и 1;
- `ChgCash` — право на изменение состояния счета абонента, может принимать значения 0 и 1;
- `ChgPassword` — право на изменение пароля абонента, может принимать значения 0 и 1;
- `UsrAddDel` — право на добавление и удаление абонентов, может принимать значения 0 и 1;
- `ChgAdmin` — право на изменение параметров администраторов системы а также их добавление и удаление, может принимать значения 0 и 1;
- `ChgTariff` — право на добавление, удаление и изменение тарифов, может принимать значения 0 и 1.

11.2 Описание формата файлов `/var/stargazer/users/*/*`.

Данные абонентов хранятся в каталогах вида `/var/stargazer/users/<login>`, где `<login>` — логин абонента. В каталогах содержатся файлы `conf`, `stat`, `stat.<YYYY>.<MM>`, `stat2.<YYYY>.<MM>`, `log`, `log2` и подкаталоги `messages` и `detail_stat`. Файл `conf` содержит данные учетной записи абонента, файл `stat` содержит статистику абонента за последний месяц, файлы `stat.<YYYY>.<MM>` и `stat2.<YYYY>.<MM>` содержат статистику абонента за предыдущие месяцы. Они различаются только полнотой представления — файл `stat2.<YYYY>.<MM>` содержит больше информации. Файлы `log` и `log2` содержат журнал изменений (в том числе подключений и отключений). Файл `log2` идентичен файлу `log` за исключением того что при отключении абонента в него пишется причина отключения. Подкаталог `messages` содержит файлы с сообщениями абоненту. Имена файлов представляют собой метку времени создания сообщения в формате UNIX Timestamp.

Подкаталог `detail_stat` содержит в себе детальную статистику абонента, разбитую по годам, месяцам и дням: `<YYYY>/<MM>/<DD>`. Файлы учетной записи, статистики и детальной статистики представлены в формате «ключ-значение». Каждая строка файла состоит из названия параметра и его значения разделенных символом «=», пустые строки и строки начинающиеся с символа «#» (комментарии) игнорируются.

Описание параметров файла `conf`:

- `Password` — пароль абонента (в открытом виде);
- `IP` — список IP-адресов, разделенных запятой, с которых разрешена авторизация абонента (может содержать * — авторизация разрешена с любого адреса);
- `Tariff` — название текущего тарифного плана абонента;
- `TariffChange` — название тарифного плана, который будет установлен в начале следующего месяца (может быть пустым);
- `Credit` — величина кредита абоненту;
- `CreditExpire` — дата истечения срока кредитования абонента в формате UNIX Timestamp (может содержать 0 — «вечный» кредит);
- `DisabledDetailStat` — параметр, отключающий ведение детальной статистики для абонента, может принимать значения 0 и 1;
- `Down` — принудительное отключение абонента, может принимать значения 0 и 1;
- `Passive` — «заморозка» абонента (для таких абонентов запрещена авторизация и не снимается абонентская плата), может принимать значения 0 и 1;
- `RealName` — реальные ФИО абонента, текстовое поле;
- `Address` — почтовый адрес абонента в произвольном формате, текстовое поле;
- `Phone` — номера телефонов абонентов в произвольном формате, текстовое поле;
- `Note` — примечание, текстовое поле;
- `Email` — адрес электронной почты абонента, текстовое поле;
- `Group` — группа в которую входит абонент, текстовое поле;
- `AlwaysOnline` — включение доступа без авторизации, может принимать значение 0 или 1, для функционирования требует плагин `auth_ao`;
- `CreationTime` — момент создания учетной записи абонента, UNIX Timestamp;
- `Userdata0 ... Userdata9` — дополнительные текстовые параметры которые могут содержать вспомогательную информацию.

Описание параметров файла `stat`:

- `U0 ... U9, D0 ... D9` — текущие данные о трафике абонента в байтах по направлениям;
 - `Cash` — состояние счета абонента;
 - `FreeMb` — количество «бесплатных мегабайт» у абонента (в деньгах);
 - `LastActivityTime` — момент последней активности абонента, UNIX Timestamp;
 - `LastCashAdd` — последнее изменение счета абонента;
 - `LastCashAddTime` — момент последнего изменения счета абонента, UNIX Timestamp;
-

- `PassiveTime` — количество секунд проведенное в состоянии «заморожен».

Описание параметров файла `stat.<YYYY>.<MM>`

- `U0 ... U9, D0 ... D9` — данные о трафике абонента в байтах по направлениям на конец месяца;
- `Cash` — состояние счета абонента на конец месяца;

Описание параметров файла `stat2.<YYYY>.<MM>`

- `U0 ... U9, D0 ... D9` — данные о трафике абонента в байтах по направлениям на конец месяца;
- `Cash` — состояние счета абонента на конец месяца;
- `FreeMb` — количество «бесплатных мегабайт» у абонента (в деньгах) на конец месяца;
- `LastActivityTime` — момент последней активности абонента, UNIX Timestamp;
- `LastCashAdd` — последнее изменение счета абонента;
- `LastCashAddTime` — момент последнего изменения счета абонента, UNIX Timestamp;
- `PassiveTime` — количество секунд проведенное в состоянии «заморожен».

11.3 Описание формата файлов `/var/stargazer/tariff/*.tf`.

Файлы `/var/stargazer/tariffs/*.tf` содержат информацию о тарифах. Имя файла имеет вид `<name>.tf`, где `<name>` — название тарифа. Файл имеет формат «ключ-значение». Каждая строка файла содержит название параметра и его значение разделенные символом «=», пустые строки и строки начинающиеся с символа «#» (комментарии) игнорируются.

Описание тарифа состоит из общих параметров и параметров специфичных для каждого направления. Параметры описывающие направления содержат в своем названии номер направления к которому они относятся.

Описание параметров файла тарифов:

- `Fee` — величина абонентской платы;
- `Free` — количество «бесплатных мегабайт» входящих в абонентскую плату (в деньгах);
- `PassiveCost` — стоимость «заморозки» абонента;
- `TraffType` — тип тарификации трафика:
 - `up+down` — входящий и исходящий трафик тарифицируется суммарно;
 - `up` — тарифицируется только исходящий трафик;
 - `down` — тарифицируется только входящий трафик;
 - `max` — тарифицируется большее количество трафика;
- `PriceDayA0 ... PriceDayA9` — цена трафика по соответствующему направлению в дневной период до превышения порога;
- `PriceDayB0 ... PriceDayB9` — цена трафика по соответствующему направлению в дневной период после превышения порога;
- `PriceNightA0 ... PriceNightA9` — цена трафика по соответствующему направлению в ночной период до превышения порога;

- PriceNightB0 ... PriceNightB9 — цена трафика по соответствующему направлению в ночной период после превышения порога;
 - Threshold0 ... Threshold9 — величина порога по каждому из направлений в байтах;
 - Time0 ... Time9 — интервал времени считающийся «дневным» (остальное время считается «ночным») в формате HH1:MM1-HH2:MM2;
 - SinglePrice0 ... SinglePrice9 — отключает «пороговую» тарификацию (трафик при этом считается по допороговой цене), может принимать значения 0 и 1;
 - NoDiscount0 ... NoDiscount9 — отключает тарификацию по времени (трафик при этом считается по «дневной» цене), может принимать значения 0 и 1.
-

Глава 12

Приложение В

12.1 Описание процесса ручной установки БД для СУБД Firebird

Для работы модуля необходимо завести учетную запись пользователя СУБД Firebird. Для этого используется утилита управления учетными записями gsec. Пример создания пользователя (подробности: Server configuration and management):

```
# gsec -user sysdba -password masterkey
GSEC> add stg -pw 123456
```

Для корректной работы модуля пользователь должен иметь право на запись и чтение данных в таблицы БД.

Модуль для работы с СУБД Firebird требует наличия правильно созданной базы данных. SQL-скрипт для создания находится в каталоге projects/stargazer/inst/var/00-base-00.sql. В обычных условиях выполнение make install или make install-data автоматически создает базу данных, используя этот скрипт. Для того чтобы создать ее вручную необходимо в файле SQL-скрипта раскомментировать строки:

```
connect 'localhost:/var/stg/stargazer.fdb' user 'stg' password '123456';
drop database;
create database 'localhost:/var/stg/stargazer.fdb' user 'stg' password '123456' default ↵
character set win1251;
```

и заменить в них адрес сервера, путь к базе, имя пользователя БД и пароль. После этого выполнить команду isql -i <файл_скрипта>.

Внимание! Для пользователей Debian и, возможно, некоторых других дистрибутивов командный интерпретатор SQL для СУБД Firebird называется isql-fb.

12.2 Описание структуры БД для СУБД Firebird

Таблица tb_admins (учетные записи администраторов системы):

- pk_admin – уникальный идентификатор учетной записи администратора;
 - login – название учетной записи администратора;
 - passwd – пароль для учетной записи администратора (в зашифрованном виде);
 - chg_conf – флаг, позволяющий менять конфигурационные параметры пользователя;
-

- `chg_password` – флаг, позволяющий менять пароль пользователя;
- `chg_stat` – флаг, позволяющий менять статистические данные пользователя;
- `chg_cash` – флаг, позволяющий менять количество денег на счету у пользователя;
- `usr_add_del` – флаг, позволяющий удалять и добавлять пользователей;
- `chg_tariff` – флаг, позволяющий управлять тарифами;
- `chg_admin` – флаг, позволяющий управлять учетными записями администраторов системы;
- `chg_service` – зарезервировано;
- `chg_corporation` – зарезервировано.

С таблицей связан триггер `tr_admin_bi`, выполняющий автогенерацию идентификатора (первичного ключа) учетных записей администраторов системы.

Таблица `tb_tariffs` (описание тарифов):

- `pk_tariff` – уникальный идентификатор тарифа;
- `name` – название тарифа;
- `fee` – абонплата;
- `free` – количество бесплатных мегабайт трафика, входящих в абонплату;
- `passive_cost` – стоимость «заморозки» пользователя;
- `traf_type` – тип подсчета трафика (`upload`, `download`, `upload+download`, `MAX(upload, download)`).

С таблицей связана хранимая процедура `sp_add_tariff`, получающая в качестве параметров название тарифа и количество направлений. Возвращает сгенерированный идентификатор (первичный ключ) тарифа. При вызове создает запись в таблице `tb_tariffs` и указанное количество записей в таблице `tb_tariffs_params`, соответствующее количеству направлений. С таблицей так же связана хранимая процедура `sp_delete_tariff`, получающая на вход название тарифа и удаляющая его из таблицы `tb_tariffs` и связанные с ним данные из таблицы `tb_tariffs_params`.

Таблица `tb_tariffs_params` (параметры тарифов):

- `pk_tariff_param` – уникальный идентификатор записи информации о направлении в тарифе;
 - `fk_tariff` – внешний ключ для связи с тарифом;
 - `dir_num` – номер направления;
 - `price_day_a` – стоимость 1 Мб трафика днем до превышения порога;
 - `price_day_b` – стоимость 1 Мб трафика днем после превышения порога;
 - `price_night_a` – стоимость 1 Мб трафика ночью до превышения порога;
 - `price_night_b` – стоимость 1 Мб трафика ночью после превышения порога;
 - `threshold` – порог по трафику;
 - `time_day_begins` – время начала дня;
 - `time_day_ends` – время конца дня.
-

С таблицей связан триггер `tr_tariffs_params_bi`, выполняющий автогенерацию идентификаторов записей (первичных ключей) в таблице.

Таблица `tb_corporations` – зарезервирована.

С таблицей связан триггер `tr_corporations_bi`, выполняющий автогенерацию первичных ключей.

Таблица `tb_users` (учетные записи пользователей системы):

- `pk_user` – уникальный идентификатор пользователя;
- `fk_tariff` – внешний ключ для связи с тарифом;
- `fk_tariff_change` – внешний ключ для связи с тарифом (может быть пустым);
- `fk_corporation` – зарезервировано;
- `address` – адрес пользователя;
- `always_online` – флаг «всегда on-line»;
- `credit` – величина кредита в деньгах;
- `credit_expire` – дата окончания действия кредита;
- `disabled_detail_stat` — отключение детальной статистики пользователя;
- `disabled` – флаг «отключен»;
- `email` – email пользователя;
- `grp` – название группы пользователя;
- `note` – примечание;
- `passive` – флаг «заморозки»;
- `passwd` – пароль пользователя (в открытом виде);
- `phone` – номер телефона пользователя;
- `name` – название учетной записи пользователя (логин);
- `real_name` – реальное имя пользователя.

С таблицей связана хранимая процедура `sp_add_user`, принимающая на вход 2 параметра: название учетной записи пользователя и количество направлений. Возвращает сгенерированное значение первичного ключа. При вызове создает запись в таблице `tb_users`, соответствующую ей запись в таблице `tb_stats` и указанное количество записей в таблице `tb_stats_traffic`. Так же с таблицей связана хранимая процедура `sp_delete_user`, получающая в качестве входного параметра название учетной записи (логин) пользователя и удаляющая учетную запись и всю сопутствующую информацию (данные пользователя, IP-адреса, сообщения, журналы параметров, статистику и детальную статистику, журналы сессий) о пользователе из базы.

Таблица `tb_detail_stats` (детальная статистика по пользователям системы):

- `pk_detail_stat` – уникальный идентификатор записи детальной статистики;
 - `fk_user` – внешний ключ для связи с пользователем;
 - `dir_num` – номер направления;
 - `ip` – удаленный IP-адрес (в формате 32-битного беззнакового целого);
 - `download` – количество принятых байт;
-

- `upload` – количество отправленных байт;
- `cost` – стоимость объема данных;
- `from_time` – время начала сессии;
- `till_time` – время завершения сеанса.

С таблицей связан триггер `tr_detail_stats_bi`, выполняющий автогенерацию уникальных идентификаторов записей детальной статистики (первичных ключей).

Таблица `tb_services` – зарезервирована. С ней связан триггер `tr_services_bi`, выполняющий автогенерацию первичных ключей. Так же с ней связана хранимая процедура `sp_delete_service`, удаляющая всю информацию связанную с записью в данной таблице.

Таблица `tb_users_services` – зарезервирована. С ней связан триггер `tr_users_services_bi`, выполняющий автогенерацию первичных ключей.

Таблица `tb_messages` (сообщения пользователям):

- `pk_message` – уникальный идентификатор сообщения;
- `fk_user` – внешний ключ для связи с пользователем;
- `ver` – версия формата сообщения;
- `msg_type` – тип сообщения;
- `last_send_time` – время последней отсылки сообщения;
- `creation_time` – время создания сообщения;
- `show_time` – время показа сообщения;
- `repeat` – количество повторений;
- `repeat_period` – период повторения;
- `msg_text` – текст сообщения.

С таблицей связана хранимая процедура `sp_add_message`, которая получает на вход идентификатор, параметры и текст сообщения. В случае если идентификатор пустой – сообщение добавляется в таблицу. Если не пустой – параметры и текст сообщения обновляются. Процедура возвращает идентификатор сообщения.

Таблица `tb_stats` (статистика пользователя):

- `pk_stat` – уникальный идентификатор записи статистики;
 - `fk_user` – внешний ключ для связи с пользователем;
 - `cash` – количество денег на счету;
 - `free_mb` – количество не истраченных бесплатных Мб;
 - `last_activity_time` – время последней активности пользователя;
 - `last_cash_add` – последнее пополнение счета;
 - `last_cash_add_time` – время последнего пополнения счета;
 - `passive_time` – время, проведенное пользователем в «замороженном» состоянии за месяц (используется для вычисления объема снимаемой абонплаты);
 - `stats_date` – дата записи статистики (месячной статистики).
-

С таблицей связана хранимая процедура `sp_add_stat`, получающая на вход набор данных статистики, добавляющая запись в таблицу и возвращающая сгенерированный первичный ключ записи.

Таблица `tb_stats_traffic` (статистика по направлениям):

- `pk_stat_traffic` – уникальный идентификатор записи статистики;
- `fk_stat` – внешний ключ для связи со статистикой пользователя (текущей или месячной);
- `dir_num` – номер направления;
- `download` – объем входящего трафика;
- `upload` – объем исходящего трафика.

С таблицей связан триггер `tr_stat_traffic_bi`, служащий для автогенерации первичного ключа.

Таблица `tb_users_data` (данные пользователя):

- `pk_user_data` – уникальный идентификатор записи;
- `fk_user` – внешний ключ для связи с пользователем;
- `num` – номер поля данных;
- `data` – текст поля данных.

С таблицей связан триггер `tr_user_data_bi`, служащий для автогенерации первичного ключа.

Таблица `tb_allowed_ip` (IP-адреса пользователя):

- `pk_allowed_ip` – уникальный идентификатор записи;
- `fk_user` – внешний ключ для связи с пользователем;
- `ip` – IP-адрес (в формате 32-битного беззнакового целого);
- `mask` – маска (в формате 32-битного беззнакового целого).

С таблицей связан триггер `tr_allowed_ip_bi`, служащий для автогенерации первичного ключа.

Таблица `tb_sessions_log` (журнал сессии):

- `pk_session_log` – уникальный идентификатор записи в журнале;
- `fk_user` – внешний ключ для связи с пользователем;
- `event_time` – время события;
- `event_type` – тип события (`connect/disconnect`);
- `ip` – IP-адрес пользователя.

С таблицей связана хранимая процедура `sp_append_session_log`, получающая в качестве параметров данные для записи в журнал сессии. Добавляет запись в журнал и возвращает сгенерированный первичный ключ.

Таблица `tb_sessions_data` (данные сессии):

- `pk_session_data` – уникальный идентификатор записи данных;
 - `fk_session_log` – внешний ключ для связи с записью в журнале сессии;
-

- `dir_num` – номер направления;
- `session_upload` – объем исходящего трафика за сессию;
- `session_download` – объем входящего трафика за сессию;
- `month_upload` – объем исходящего трафика за месяц;
- `month_download` – объем входящего трафика за месяц.

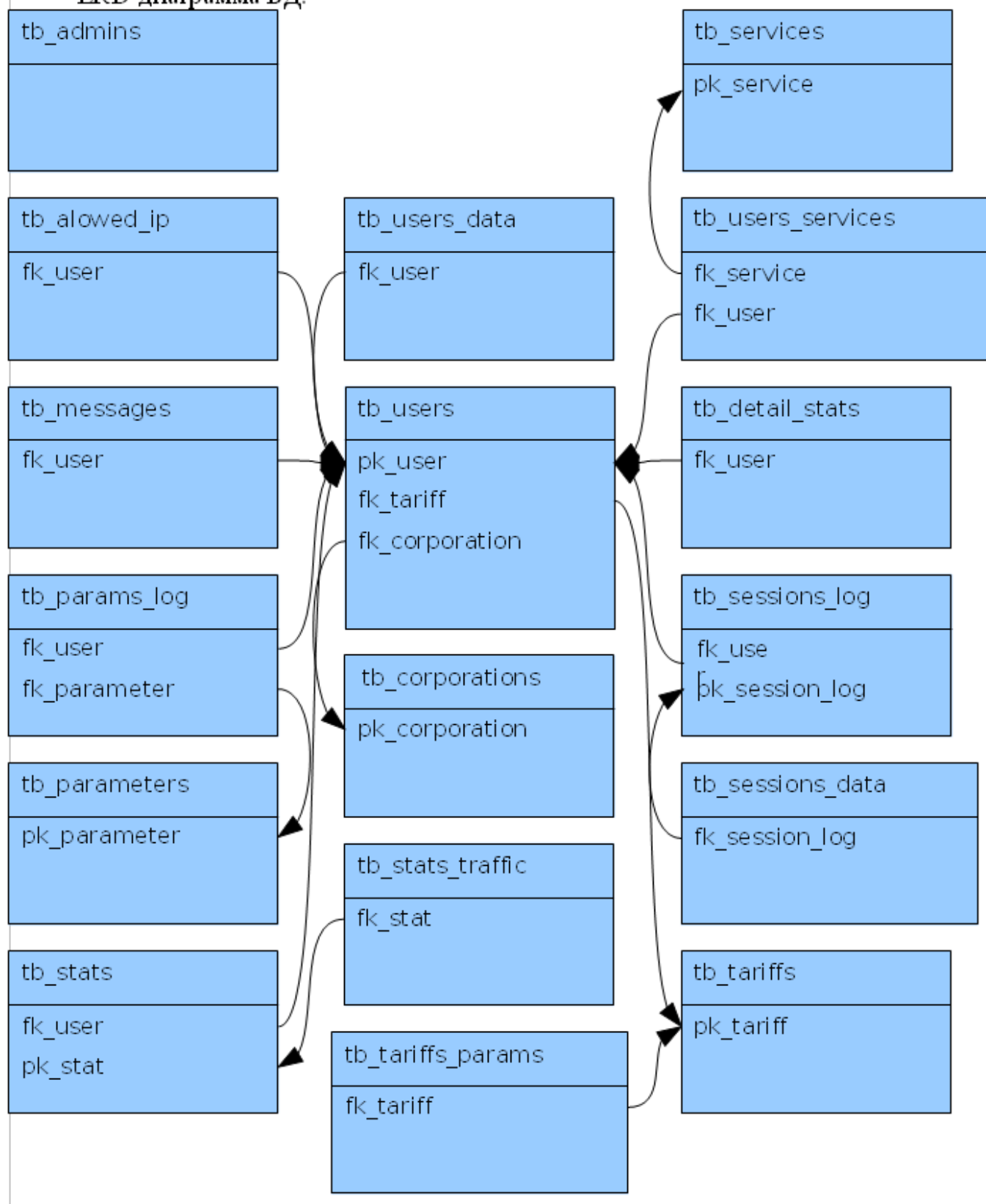
С таблицей связан триггер `tr_session_data_bi`, служащий для автогенерации первичного ключа. Таблица `tb_parameters` (параметры пользователя):

- `pk_parameter` – уникальный идентификатор параметра;
- `name` – название параметра.

С таблицей связан триггер `tr_parameter_bi`, служащий для автогенерации первичного ключа. Таблица `tb_params_log` (журнал изменений параметров пользователя):

- `pk_param_log` – уникальный идентификатор записи в журнале изменений параметров пользователя;
- `fk_user` – внешний ключ для связи с пользователем;
- `fk_parameter` – внешний ключ для связи с таблицей параметров;
- `event_time` – время изменения;
- `from_val` – старое значение (в виде строки);
- `to_val` – новое значение (в виде строки);
- `comment` – комментарий.

С таблицей связан триггер `tr_param_log_bi`, служащий для автогенерации первичного ключа.

ERD-диаграмма БД:

Глава 13

Приложение Г

13.1 Описание процесса установки БД для СУБД MySQL

Для работы модуля необходимо завести учетную запись пользователя СУБД MySQL. Для этого можно использовать командный интерпретатор mysql:

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9 to server version: 5.0.26-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create user stg_user identified by '123456';
Query OK, 0 rows affected (0.04 sec)
```

Модуль для работы с MySQL автоматически создает БД и необходимые для его работы таблицы. Для этого у пользователя, указанного в настройках должны быть установлены соответствующие права (см. документацию по настройке MySQL: MySQL User Account Management). Модуль может корректно работать и с пользователем с ограниченными правами (полный доступ только к своей БД). Для этого необходимо сперва вручную создать БД, а затем дать пользователю права на доступ к ней. Например так:

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9 to server version: 5.0.26-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create user stg_user identified by '123456';
Query OK, 0 rows affected (0.04 sec)

mysql> create database stg_database;
Query OK, 1 row affected (0.02 sec)

mysql> grant all on stg_database.* to stg_user;
Query OK, 0 rows affected (0.06 sec)
```

После этого модуль при первом запуске сам создаст необходимые таблицы.

13.2 Описание структуры БД для СУБД MySQL

Таблица admins (учетные записи администраторов системы):

- login – название учетной записи администратора системы;
- password – пароль для учетной записи администратора системы;
- ChgConf – флаг, разрешающий менять конфигурационные параметры пользователи;
- ChgPassword – флаг, позволяющий менять пароль пользователя;
- ChgStat – флаг, позволяющий менять статистические данные пользователя;
- ChgCash – флаг, позволяющий менять количество денег на счету у пользователя;
- UsrAddDel – флаг, позволяющий управлять учетными записями пользователей;
- ChgTariff – флаг, позволяющий управлять тарифами;
- ChgAdmin – флаг, позволяющий управлять учетными записями администраторов системы.

Таблица tariffs (описание тарифов системы):

- name – название тарифа;
- PriceDayA<n> - стоимость 1 Мб трафика по направлению <n> днем до превышения порога;
- PriceDayB<n> - стоимость 1 Мб трафика по направлению <n> днем после превышения порога;
- PriceNightA<n> - стоимость 1 Мб трафика по направлению <n> ночью до превышения порога;
- PriceNightB<n> - стоимость 1 Мб трафика по направлению <n> ночью после превышения порога;
- Threshold<n> - порог по трафику;
- Time<n> - время дня;
- NoDiscount<n> - флаг, указывающий, что трафик по направлению <n> считается без учета порогового значения;
- SinglePrice<n> - флаг, указывающий, что трафик по направлению <n> считается без учета времени суток;
- PassiveCost – стоимость «заморозки» учетной записи пользователя;
- Fee – абонплата;
- Free – количество бесплатных Мб трафика, входящих в абонплату;
- TraffType – тип подсчета трафика (up, down, up+down, max).

Таблица users (учетные записи пользователей системы и их статистика):

- login – название учетной записи пользователя;
 - Password – пароль для учетной записи пользователя;
 - Passive – флаг режима «Заморожен»;
 - Down – флаг режима «Отключен»;
 - AlwaysOnline – флаг режима «Всегда on-line»;
-

- Tariff – название тарифа;
- Address – адрес пользователя;
- Phone – номер телефона пользователя;
- Email – адрес электронной почты пользователя;
- Note – примечания;
- RealName – реальное имя пользователя;
- StgGroup – название группы пользователя;
- Credit – величина кредита в деньгах;
- TariffChange – название тарифа, на который будет переведен пользователь в начале месяца;
- Userdata<n> – поля дополнительных данных пользователя;
- CreditExpire – дата окончания действия кредита пользователя;
- DisabledDetailStat — отключение детальной статистики пользователя;
- IP – список IP-адресов пользователя;
- D<n> – объем входящего трафика по направлению <n>;
- U<n> – объем исходящего трафика по направлению <n>;
- Cash – количество денег на счету;
- FreeMb – количество бесплатных мегабайт;
- LastCashAdd – последнее пополнение счета;
- LastCashAddTime – время последнего пополнения счета;
- PassiveTime – время, проведенное пользователем в «замороженном» состоянии за месяц (используется для вычисления объема снимаемой абонплаты);
- LastActivityTime – время последней активности пользователя;
- NAS – используется совместно с модулем захвата трафика для NetFlow.

Таблица messages (сообщения пользователю):

- login – название учетной записи пользователя, для которой предназначено сообщение;
- id – уникальный идентификатор сообщения;
- type – тип сообщения;
- lastSendTime – время последней отсылки сообщения;
- creationTime – время создания сообщения;
- showTime – время показа сообщения;
- stgRepeat – количество повторений сообщения;
- repeatPeriod – период повторения сообщения;
- text – текст сообщения.

Таблица stat (месячная статистика пользователей):

- login – название учетной записи пользователя, которой принадлежит запись в статистике;
- month – месяц записи статистики;
- year – год записи статистики;
- U<n> – объем исходящего трафика за месяц по направлениям;
- D<n> – объем входящего трафика за месяц по направлениям;
- cash – количество денег на счету.

Таблица logs_<m>_<y> (помесячный журнал сессий пользователя и изменений его параметров):

- unid – уникальный идентификатор записи;
- login – название учетной записи пользователя;
- text – текст записи в журнале.

Таблица detailstat_<m>_<y> (помесячная детальная статистика пользователей):

- login – название учетной записи пользователя;
 - day – порядковый номер дня в месяце;
 - startTime – время начала сессии;
 - endTime – время завершения сессии;
 - IP – удаленный IP-адрес;
 - dir – направление;
 - up – объем исходящего трафика;
 - down – объем входящего трафика;
 - cash – стоимость трафика.
-