NetUP

ADMINISTRATOR'S GUIDE

UTM5

billing system
v. 5.3-004

Contents

1 License agreement

2 Introduction

3 System description

4 Basic system objects

5 Administrator’s interface

6 Installation and initial setup

7 Usage examples

8 UTM5 core

9 UTM5 RADIUS

10 Text files import

11 UTM5 RFW

12 UTM5 Dynashape

13 UTM5 Urfaclient

14 Structured data import

15 Dealer module

16 cashier module

17 UTM5 tray utility

18 Web interface

19 Hotspot module

20 UTM5 DHCP

21 IPTV integration module

22 IP telephony module

23 Automatic registration of users

24 E-mail notifications

25 System maintenance

26 Auxiliary utilities

27 Payment systems

28 Appendix

LICENSE AGREEMENT

This License agreement (hereinafter "the Agreement") is a legal agreement made between you (natural or legal person, hereinafter "the user") and NetUP company (hereinafter "NetUP") concerning the above software NetUP product, and including the software, print materials, and any inline or electronic documentation (hereinafter "the product").

1. License scope.
    1.1. One product set may be installed on only one computer.
    1.2. The product is considered used if it is loaded in the RAM memory, or stored on hard disk, CD or other storing device of a certain computer.
    1.3. The user has a right to create an archive copy intended exclusively for individual use for the product recovery or storing it on hard disk on condition that the original copy is stored as backup or archive copy.
2. The user rights and warranty services are rendered only to the registered users.
3. This license also refers to the original product updates and add-ons provided by NetUP, if not specified otherwise in the accompanying documents.
4. The product may be transferred to another workstation. The original user has a right to transfer it on a non-recurrent basis to another user. This transfer shall include all necessary components: carriers and print materials, this agreement and authenticity certificate (if available). It is not allowed to provide the product for rent, lease or temporary use.
5. Sharing or distributing the product or any part of it or the license key files is prohibited.
6. It is not allowed to disclose the technology or decompile the product, except in cases when despite this restriction in the license agreement it is explicitly allowed by the active legislation, and to the degree that is clearly defined therein.
7. NetUP has a right to terminate this agreement when its provisions and terms are violated by the user. In this case, the user is obliged to destroy all available product copies and its components.
8. The user agrees that NetUP has a right to collect and use at its own discretion the technical data supplied by the user to the support department of NetUP.
9. To the maximum degree allowed by the active legislation and under no circumstances, NetUP and its vendors are liable for any special, occasional, direct or indirect damage, or loss (including without limitations the following: loss of profit; loss of confidential or other information; loss caused by breakages in commercial or production operations; damage to health; violation of privacy; non-fulfillment of any obligation including obligation to act conscientiously and with reasonable precautions; loss caused by negligence; and any other property or other kind loss) in result of use or impossibility to use the product, rendering or

non-rendering maintenance services, or in other cases envisaged or related to provisions of this license agreement, including violation of obligation, civil offence (including negligence), impersonal (independent of guilt) liability for some damage, violation by NetUP or its vendor of any contract or warranty liabilities, even if NetUP or its vendors have been notified in advance of this possible damage. NetUP and its vendors are not liable and bear no obligations in case of this product unauthorized use, which is not envisaged by this agreement.

10. This agreement becomes valid on the moment of installing the product. This agreement will be valid for an indefinite time period, except cases of transferring the right for this product use, as envisaged by p. **4** of this Agreement.
11. All property rights, author's rightsfor this product and in relation to it belong to NetUP or its vendors. This product is provided for use ("licensed"), not for sale.
12. Installation and maintenance of the product is provided by NetUP under a separate agreement.

3

INTRODUCTION

Intended audience

The present manual is intended for the providers of Internet and telephony services using the NetUP UTM5 billing system. The document covers basic functionality of UTM5 v. 5.3-001, as well as the typical necessary actions during its startup, deployment, and operation.

Notation conventions

The following elements are emphasized in the text:

Terms (on first occurrence)

Interface strings

Hyperlinks

`Inline code pieces and commands`

```
Code blocks
```

Notes

Warnings related to incomplete compatibility with older versions of UTM5

Generic warnings

List of terms

Networking

- **DNS –** (Domain Name Service) is a distributed system that translates human-readable domain names to numerical IP addresses.
- **TCP/IP –** (Transmission Control Protocol / Internet Protocol) is a stack of protocols used in computer networks.
- **HTTP –** (Hypertext Transport Protocol) is an application layer network protocol.
- **HTTPS –** (HTTP Secure) is extension of HTTP enabling encrypted communication.

- **UDP –** (User Datagram Protocol) is a transport level network protocol used to transfer data without guaranteed reliability.
- **TCP –** (Transmission Control Protocol) is a transport level network protocol used for reliable and ordered transfer of data.
- **MAC –** (Media Access Control) is a level of communications system that provides addressing and channel access control mechanisms.
- **MAC address –** is a unique identifier assigned to network interfaces for communications on the physical network segment.
- **SSL –** (Secure Socket Layer) is a cryptographic network protocol.
- **Hub –** is a networking device that connects devices within a segment.
- **Switch –** is a networking device that connects network segments.

Other terms

- **XML –** (eXtensible Markup Language) is a language for representing structured data.
- **Database –** is a system that organizes and stores large amounts of data.
- **Cluster –** is a group of connected computers used as a single computing facility.
- **Billing system –** is a system that automatically accounts for the use of services and sends bills to the customers.
- **ISO –** is the International Organization for Standardization.
- **ANSI –** is the American National Standards Institute.

SYSTEM DESCRIPTION

Main features of the system

NetUP UTM5 billing system (automated payment system) is a professional solution intended for automated settling of telecommunication providers with subscribers for services provided. Basic module of the system supports accounting for leased lines. Additionally, the system allows creating and keeping of records of periodical or one-time services. With the use of extra modules the system calculates VPN and PPPoE connections, IP telephony services, dial-up access, and network wireless access (hotspot).

Main entities of the system (users, services, tariff plans, etc.) together with their characteristics are listed in **Basic system objects** on page **25**. Administrator's interface is described in **Administrator's interface** on page **41**. Working examples are presented in **Usage examples** on page **129**.

Main mechanism of processing statistical data on consumed traffic put in the system allows to process simultaneously an unlimited number of communication paths.

NetUP UTM5 supports keeping of database for customers, banks, routers, firewalls, IP address zones, houses connected, services provided, etc.

The current version of the UTM billing system was created basing on considerable experience of introducing previous versions and users' requests. For more convenient work of the program the administrator control center was implemented as Java GUI application able to run on any platform.

The system fully supports work with prepaid cards. There is a possibility to export generated cards into external file of XML format. The delivered system supports Russian and English languages, but may be translated into any other language as well, if necessary. The system is able to work with several currencies at the same time.

The system can be used for generating of accounting records and various statements, as well as for keeping contracts database. For more convenient customers technical support the system has a message exchange service.

The system blocks access to services when necessary, i.e. when the subscriber's personal account runs out of money.

The user interface (User Virtual Office) based on web technologies provides subscribers with access to their accounts and checking balance from any part of the world via the Internet. XSLT technology and patterns used for creating the User Virtual Office give the system administrators a way to change the interface independently without interrupting its functionality.

The concept of traffic class allows monitoring traffics of different networks, i.e. to tell apart the domestic and foreign traffic, or the peering and local traffic. Traffic classes may be split by various characteristics, e.g. source and recipient networks, source and recipient ports, type of service (TOS), protocol, source and recipient autonomous systems, router interface via which the packet goes, etc.

UTM5 does not support the recalculation of the already considered traffic data. If some data have been accounted for with erroneous price, it is recommended to rectify the situation with corrective payments.

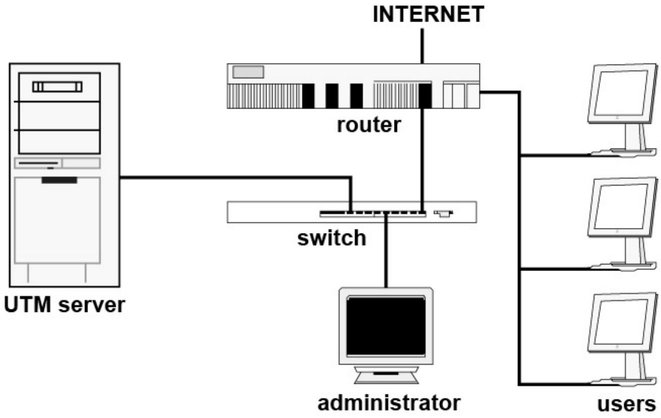
The server part of the billing system (the core of the system) is a multithreaded optimized application providing high performance of the whole system. It is especially important for networks with large client base that consumes huge amounts of traffic.

How to Connect the System to the Network

A versatile nature of the billing system allows integrating it into existing or intended network infrastructure in various ways. UTM5 can work with various hardware- and software-based routers and it doesn't limit the number of communication paths (accounted for simultaneously) and types of devices arranging these paths. Hereinafter the main commonly used variants are examined.

- Local network is connected to the Internet via a hardware-based router supporting statistics collection

Cisco, MikroTik, NSG, Revolution and other routers typically support export of the traffic statistics. In that case the server with the billing system may be installed either inside or outside the local network (e.g. in the head office available via Internet). Statistics is collected and routers are monitored remotely.



- Local network is connected to the Internet via switch or hardware-based router not supporting statistics collection.

In this case the server is installed into a segment of the local network so that traffic is available for monitoring on the server at the IP packets level. E.g., a hub is installed between the switch and the local network, and the server is connected to its port. When on, the server traps all packets going from the local network to the switch and back, analyzes their headers and processes the acquired information.

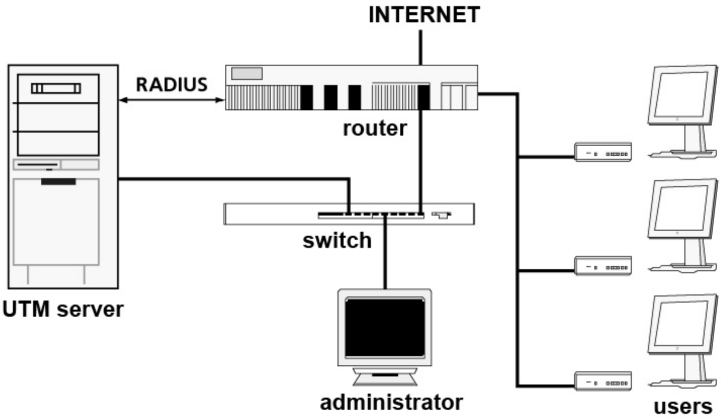
- Local network is connected to the Internet via software-based router (PC router)

This type of connection allows installing the billing system on a router or on a remote server. Statistics is read from a router interface and processed by a local machine (in the former case), or it is transmitted via the network and processed by another server (in the latter case).

Alongside with the above mentioned ways of local networks' traffic counting, there are many other ways, e.g. counting of uplinks, or any combination of the presented ways.

- Client connects to the Internet via dial-up

In this case the access server may be either Cisco or a PC-router with connected modems. Authorization of a user is performed via RADIUS protocol. Tariffication is based either on connection time or on traffic.



- Client connects to the Internet using Wi-Fi

The system supports accounting for services of wireless access using the Wi-Fi technology also known as Hotspot. This way of connection is often used in public places like hotels, cafeteria, or airports.

Structure of the UTM Billing System

The UTM billing system is a bunch of applications which mainly consists of three groups: the core of the system, the administrator interface and the user interface. The core of the system is launched on the server and is responsible for functioning of the billing system as a whole. It is described in more details in **UTM5 core** on page **145**. The administrator interface (see **Administrator's interface** on page **41**) is a Java application installed on the administrator's workstation and allowing to fine-tune and control the system. This application is platform independent and can be run on any operating system, including Windows, Linux or FreeBSD. The user interface is a set of programs working along with the web server and constituting the user's virtual office.

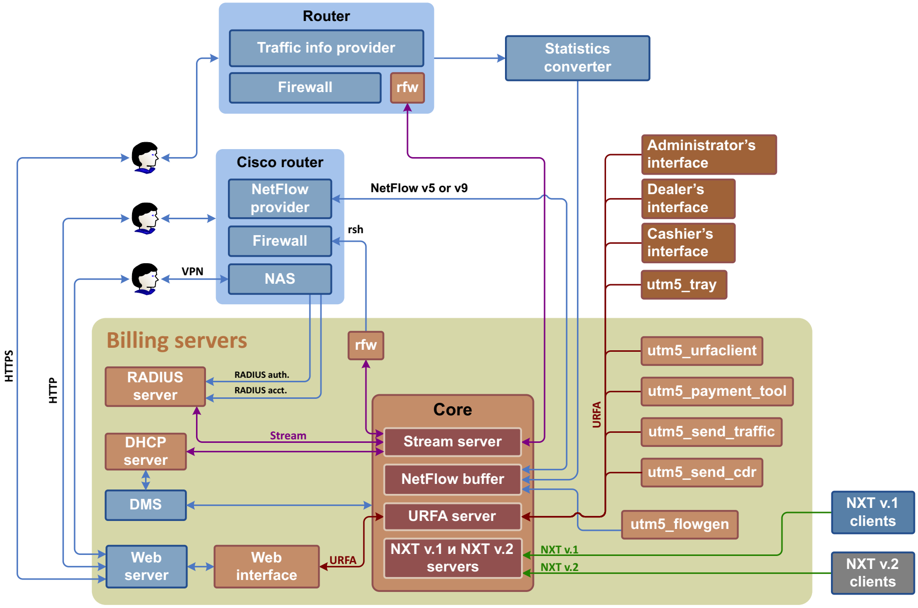


Рис. 1. UTM5 schematics

UTM Remote Function Access (URFA)

URFA is a module for access to the core of the system from external applications. It authorizes users according to CHAP scheme and provides the work of a remote user. The protocol supports data transmission and function calls. URFA checks up whether a certain user has access to the function called, and on positive check allows the user to start the data exchange. Otherwise the access is rejected.

Each session is given a 128-bit replication-free system ID number (SID). SID can be used repeatedly to gain access to the system. In case of transient error (when a session is being restored) SID is removed, and the user is prompted to enter login and password once more. SID is related to the user’s IP address, and is removed automatically after some downtime (see the `web_session_timeout` variable). An option of session restore is available, which requires the system user’s rights.

After the session has been started, the table of permitted calls is created including a list of access rights and symbols generated by the system. If, after the session has been started up, an additional module is loaded, the corresponding calls will be listed as forbidden. In that case the user must log in once more.

User's rights

Users of the system are split into two types: ordinal users (clients, subscribers) and administrators (system users). Depending on the type, a user has some list of permitted operations. The list of permitted operations can be checked up (and altered) in **System groups** on page **28**. The clients are permitted to execute operations with the identifier more than `0x80000000`; other operations are accessed by administrators only.

Version policy

Every new major release of UTM5 is preceded by the release candidate version(s), which are suitable for early feature testing but not recommended to use in production. These are followed by the release version. If any critical problems are discovered later, one or more update versions may be issued; these contain no new functionality as compared to the release version, only the bug fixes. It is recommended to install the latest update, or just the latest release if there are no updates to it.

Logging

If some system component needs to leave a log message, it passes the event level and message text to the logging module.

Following event levels exist:

Level	Name	Description
0	EMERG	Fatal error, system halted
1	ALERT	Critical error, requires immediate action
2	CRIT	Critical error
3	ERROR	Non-critical error
4	Warn	Warning
5	Notice	Information that may worth noticing
6	Info	General information
7	Debug	Debugging information

Level	Name	Description
8	Trace	Additional debugging information
9	Stats	Statistics

The logging module puts the text to the appropriate log stream, depending on the module settings and the event level. The stream is associated with certain file also specified in the module settings. By default all streams are associated with the standard error stream.

There are following log streams:

Stream	Levels included
Critical	From 0 to 2
Main	From 0 to (3 + `log_level`)
Debugging	All

Some components may activate the built-in mechanism of log file rotation. At that, after logging an event the module checks file size against some specified threshold. If the file size exceeds the threshold, the file is closed and renamed to include a certain suffix. Namely, the suffix is "`.<timestamp>`" (Unix Time Stamp of file closing time) if the number of files is unlimited, or "`.0`" or the next unused figure otherwise. After that the number of files is also checked and older files are removed if the number exceeds the limit.

Logging settings for each particular module are described in more detail later on.

External charges

UTM5 contains an integration module intended to work with Rentsoft system, which is a distributor of software and digital content. Services provided by Rentsoft are not included in the list of UTM5 services. The corresponding charges, however, are registered and gathered in a special report, see **Administrator’s interface: Custom charges report** on page **81**. The invoices for these charges are issued immediately upon the charge-off.

For more details on the possible settings and parameters of the integration module, see **http://rentsoft.ru/provider/new/netup_netup300/** (free registration required).

BASIC SYSTEM OBJECTS

Introduction

This chapter lists the basic system objects and their relations.

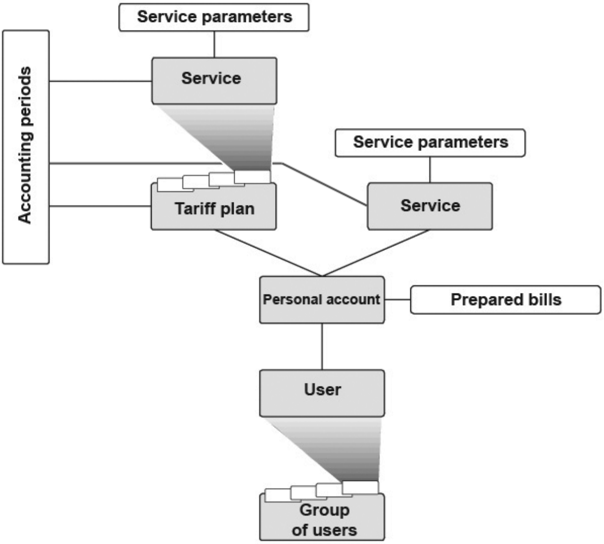


Рис. 1. Relations of the basic system objects.

Basic system objects may be split in two groups. First group is composed of the objects having immediate effect on the system’s functioning, and includes:

- **Traffic classes** (page **28**);
- **Telephone zones**;
- **Telephone directions**;
- **Time ranges** (page **37**).

Second group contains the objects which affect the system indirectly, including:

- **Tariff plans** (page **30**);
- **Accounting periods** (page **29**);
- **Services** (page **31**);
- **Currencies** (page **38**).

To clarify the distinction further: a traffic class has immediate effect on the system, since it is used non-stop in the classification of traffic data, as long as the system receives those data. On the other hand, currency has no direct involvement in the everyday routine activities and is used only to prepare invoices or accept payments, thus being an object of indirect effect.

Users

Users are the customers of services (IP traffic, telephony, etc.) controlled by the UTM5 billing system. A lot of operations may be done with the users, including: assignment of services of different types, billing, making payments, viewing statistics, etc.

Users may be combined in *groups*. Each user has one or more *accounts*, to which the services of various types may be attached, either separately or in bundles as a part of a *tariff plan*.

There is a special variety of users called *card users*. Those are created via the activation of prepaid cards (see **Prepaid cards** on page **28**).

Interface for handling users is described in **Administrator’s interface: Users** on page **44**. Group operations interface is described in **Administrator’s interface: Groups** on page **49**. Outlines of creation of users and other operations with them are given in **Usage examples: Creating users** on page **133** and the subsequent examples.

Accounts

An account is the system object containing the financial information. Account may be linked to services in a form of service links or tariff links.

Blocking

Account may get blocked, which suspends all services attached to it. The blocking may be either system (imposed automatically by the system on running out of money or on exceeding the quota), or administrative, i.e. imposed manually by the administrator.

The flags ***Don’t charge recurrent fee*** *and* ***Decrease prepaid traffic*** *and the corresponding properties related to the system behavior in respect to the blocking, which were associated with the account in UTM versions up to 5.2.1-008, have been since moved to service link properties, see* ***Service link*** *on page* ***109****.*

Below are the possible blocking types:

Type	Meaning
0	Account is not blocked
16	System blocking

Type	Meaning
48 80 112	System blocking (deprecated flags)
256	Administrative blocking
768 1280 1792	Administrative blocking (deprecated flags)
4112	System blocking on quota
4144 4176 4208	System blocking (deprecated flags)

Note that the Internet status of an account is always switched to ***Off*** *upon blocking, but not always switched automatically back to* ***On*** *upon unblocking. After lifting a manual blocking, you have to turn Internet on for that account, otherwise it will remain shut down till the next payment or start of the month, whichever comes sooner.*

Interface for handling accounts is located on the user properties page (see **Tariffication** on page **46**). Creation of accounts and other operations with them are described in **Creating account** on page **134** and subsequent examples.

System users

System users a special class of users having the rights to administrate the system using the UTM control center. System users have negative user ID. An ordinary user can not be an administrator at the same time, and vice versa.

By default, the following system users exist in the system:

- **init –** is the top-level administrator;
- **web –** is the system account for the web interface;
- **radius –** is the system account for the RADIUS server.
- **rfw –** is the system account for the RFW daemon.

The characteristic properties of a system user include: login, password, subnet the user is allowed to login from, and list of system groups to which the user belongs.

Interface for handling system users is described in **Administrator's interface: System users** on page **49**.

System groups

The access rights of a system user are determined by the *system group* where the user belongs. For simultaneous members of multiple groups, the rights are summarized among all groups. All calls for forbidden operations are registered in the system core journal.

Initially the following system groups exist in the system:

- **Wheel –** is the group of administrators (all functions permitted).
- **Dealers –** may create users, assign services and make payments.

Interface for handling system groups is described in **Administrator’s interface: System groups** on page **50**.

Prepaid cards

UTM5 may work with *prepaid cards* intended for activation via web interface (see **Web interface** on page **239**) or via the tray application (see **UTM5 tray utility** on page **233**). A card may have either limited term of use, or an expiration date.

If the card is activated on the entrance page, a card user is generated by the system. User’s login is set to `card_NUM`, where `card_` is the value of system parameter `card_user_prefix` (see **UTM5 core: Interface parameters** on page **152**) and `NUM` is the card number. User’s balance is set to the card balance value. If the card has a tariff plan attached to it, the services from the plan will be attached to the user’s personal account. If the card has limited term of use, its balance goes to the user’s account in a form of expiring payment (see **Payments** on page **38**) with this term of expiration.

Otherwise, i.e. if the card is activated by an already existing user, the card’s balance is added to the user’s account, and the tariff plan associated with the card (if any) is ignored.

Interface for handling users is described in **Administrator’s interface: Card users** on page **48**. Creation of prepaid cards is described in **Card pools** on page **51**.

Traffic classes

Traffic running towards a user and back is divided to several *classes*. Certain rules specify a set of attributes combining traffic records into classes. Traffic classes may be defined using any NetFlow v5 attributes. They are: affiliation of source or recipient's IP address with any subnet, port of source or recipient, autonomous system of source or recipient, network protocol, next router (hop), TOS, TCP flags, router interface via which a packet goes. Additionally, a traffic record may be related to a certain class depending on time and date.

Traffic belongs to a class if:

- it belongs to one of its subclasses;
- it does not belong to any its subclass with **Skip** option set;
- it came during the accounting period set for this class.

Traffic is checked against all classes in the decreasing order by ID until first match. If no match has been found, the traffic is attributed to the class with ID=0 (unclassified).

Traffic subclass is a set of features (may include the data from NetFlow records and the IP address of the NetFlow provider) that determine the attribution of the traffic to the particular class, or negation thereof.

Traffic belongs to a subclass if:

- sender and destination addresses belong to the corresponding networks set in the subclass parameters;
- the rest of NetFlow record parameters is compatible with those stated in the subclass properties;
- IP address of the NetFlow provider coincides with that set in the subclass properties, or none are set.

Interface for handling traffic classes and subclasses is described in **Administrator's interface: Traffic classes** on page **55**.

Accounting periods

An *accounting period* is a period of time to which various periodic activities are related (such as charge-off for periodic services).

Standard accounting periods are: daily, weekly, monthly, quarterly, annual, and period of fixed amount of days.

Keeping general directory of accounting periods allows settling invoices with all users or with groups of users at the same time, e.g., from the first day of a month till the first day of the next month.

When an accounting period is closed, the following operations are performed:

- recalculation of subscription fee and prepaid traffic (considering blockings);
- transfer of the prepaid traffic left (if any) to the next accounting period;
- charge-offs;
- automatic change of tariff plan, if requested;

- if Dynashape module is present (see **UTM5 Dynashape** on page **199**): issue of **Delete bandwidth limit** events for the IP addresses of service links subject to shaping, and execution of the corresponding firewall rules;
- automatic creation of a new accounting period. The new period starts exactly at the end of the one being closed, and effectively stands in its place for all purposes, i.e. has the same type, duration, number of charges, and is connected to the same service and/or tariff link(s).

Interface for handling accounting periods is described in **Administrator’s interface: Accounting periods** on page **59**.

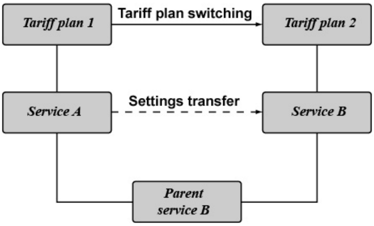
Tariff plans

A tariff plan is a bundle of services provided as a package. The system allows creating those packages, and then to assign the whole package to users at once. On attaching a tariff plan to a user it is necessary to select an accounting period and define settings for the services. A tariff plan may be set up to prolong automatically onto the next accounting period, or to switch to another compatible plan at the end of period.

Interface for working with tariff plans is described in **Administrator’s interface: Tariff plans** on page **54**. User accounts are linked to tariff plans via *tariff links* (see **Creating tariff links** on page **138**).

Tariff plans compatibility

In order to switch the tariff plan automatically the plan must be compatible with the current one. Compatible tariff plans have one-to-one correspondence between their services. This implies that the system is able to switch plans without any human intervention, all the while keeping the useful information from service links (e.g. IP addresses in IP traffic services’ settings).



Partially incompatible tariff plans can be also switched by the system, but at the cost of losing the services absent in the successor plan.

For example, consider a user having the service A attached in a tariff plan #1, and the next tariff plan set to #2, which contains the service B. In order to transmit all parameters of the service A correctly, it is necessary for the services A and B to be derived from the same parent, i.e. service template.

Services

A service is a very basic object of tariffication that defines its rules.

Interface dealing with services is described in **Administrator's interface: Services** on page **61**. Users are linked to services via *service links* (see **Creating service links** on page **136**). Service links may be created either manually (one by one), or in a bunch via tariff plans.

The sort of service determines the area where it is applicable. UTM5 supports the following sorts of services:

- Common services;
- Tariff plan services.

The type of service determines the rules of tariffication applicable to this service. UTM5 supports the following types of services:

Type	Meaning
1	One-time service
2	Periodic service
3	IP traffic service
4	Hotspot service
5	Dialup service
6	Telephony service
8	IPTV service

The common parameters for any service are:

- Service ID;
- Service name;
- Sort of service;
- Type of service.

Services of any particular type may have their specific parameters. The tariffication logic may also be type-specific.

The service costs as entered in the interface are the before-tax values. Tax rates, including the value-added tax (VAT) and sales tax, are specified separately in the user account properties and considered in all charge-offs.

All services except for **One-time services** have the start date and end date among their parameters. Start date is not used in the current release of UTM5. End date is the date when the providing of service stops, together with the charge-offs for the service. At this date the service is removed, if it is not attached to any service link.

The following types of service:

- IP traffic,
- Hotspot,
- Dialup,
- Telephony
- IPTV

have periodic portion of the cost as one of their parameters. The corresponding charge-offs are made in a similar manner to those for a periodic service, while the reports refer to them by their respective types of service.

Sorts of services

Common service

Common services are intended to be applied to some users in circumstances not provisioned by the tariff plan. For example, it may be a one-time service "Equipment setup".

A common service is:

- created directly;
- able to produce an arbitrary number of service links;
- never included in a tariff plan.

Tariff plan service

Tariff plan services are intended solely for inclusion in tariff plans. A tariff plan service is:

- never created directly (rather, it is created as a child entity to some of the existing service templates);
- always included in a tariff plan;
- able to produce only a certain number of service links, limited by the number of tariff links.

The only unique parameter that sets apart tariff plan services from common services is the **Attach by default** parameter. If this option is set, a prompt to create a service link attached to this tariff plan service will be issued on creating manually the tariff link attached to this plan. Also, on automatic creation of such a tariff link (typically, at the end of the accounting period) the service link will be created automatically as well, with all its parameters set to defaults. If **Attach by default** is not set, the service link will not be created.

Service templates

Service templates (in earlier versions of UTM5 sometimes referred to as *“fictive services”*) are used as the parent entities that produce the tariff plan services on their creation and on automatic switching of tariff plans. A service template is:

- created directly;
- not used in the tariffication logic;
- not a service by itself, i.e. is never attached to a service link;
- never included in a tariff plan, but acts as a parent to those services which are.

The interface for creation of service templates and operations with them is described in **Administrator’s interface: Service templates** on page **70**.

Ideally, there should be one service template for each logical class of services, like:

- one for the services with periodic charges;
- one for the Internet traffic using a real IP address;
- one for the general Internet traffic;
- one for the Internet users in a separated address space, etc.

Each of these service templates should have the parameters most appropriate for the particular logical class of services. These parameters will be copied by default to the derived tariff plan services, once those are created.

Types of services

One-time service

One-time service is normally intended to perform a single charge off the user’s account. The charge-off time is determined by the service link parameters. Price of service may be set to negative value, effectively turning the charge-off into a contribution. The service may also have a special parameter that requests for the exclusion of the user from some given group simultaneously with the charge-off.

Interface for creating a one-time service is described in **One-time service** on page **62**; for the corresponding service link, see **One-time service link** on page **109**.

Periodic service

Periodic service is intended for periodic charges off the user’s account. The charge-off may be applied in a variety of ways: at the beginning of an accounting period, or at the end, or in smaller portions throughout the whole period. The price to be charged in the initial period may be corrected depending on the service link parameters, and the price of the current period may depend on the user account’s parameters and/or blocking options.

Interface for creating a periodic service is described in **Periodic service** on page **62**; for the corresponding service link, see **Periodic service link** on page **110**.

IP traffic service

Services of this type are intended for tariffication of IP traffic. The price may depend on time and on the amount of traffic consumed. The service may contain so-called *prepaid traffic*, i.e. some limited amount of traffic that is passed through without payment. Also, maximum limits for traffic (so-called *quotas*) may be set up to block the user after the given amount is exhausted.

Interface for creating an IP traffic service is described in **IP traffic service** on page **64**; for the corresponding service link, see **IP traffic service link** on page **111**.

Hotspot service

Services of this type are intended to tariff hotspot access charged per time. The authorization may be done by means of RADIUS protocol (if supported by the hardware) or via the UTM5 web interface. Different prices may be set for various time ranges.

Interface for creating a hotspot service is described in **Hotspot service** on page **65**; for the corresponding service link, see **Hotspot service link** on page **114**.

Dialup service

Services of this type are intended to tariff dial-up access charged per time, possibly with different prices for various time ranges.

Interface for creating a dial-up service is described in **Dialup service** on page **66**; for the corresponding service link, see **Dialup service link** on page **114**.

Telephony service

Services of this type are intended to tariff phone calls charged per time. The call price may depend on time and call direction, and may include fixed connection price. Prepaid time option is also available. Either the caller or the called number must be registered in the properties of the telephony service link, otherwise the tariffication is impossible.

Interface for creating a telephony service is described in **Telephony service** on page **67**; for the corresponding service link, see **Telephony service link** on page **115**.

IPTV service

A services of this type is intended to tariff IP television services. It let’s one grant access to the IPTV content for a client and charge the client’s personal account with a periodic fee.

Video on demand service

A service of this type is intended to tariff video on demand (VoD) services. It allows one to temporarily grant access to VoD content and charge the client’s personal account with the rent amount.

Charge policy

The charge policy is a set of rules that are used when charging a user account. These rules are applied when a client for some reason didn’t receive the service for some part of the accounting period. This is possible when creating a service link in the middle of an accounting period (when the client will receive the service only for the rest of the accounting period), or when a client uses voluntary blocking (when the client’s account was already charged with the periodic part of the service cost).

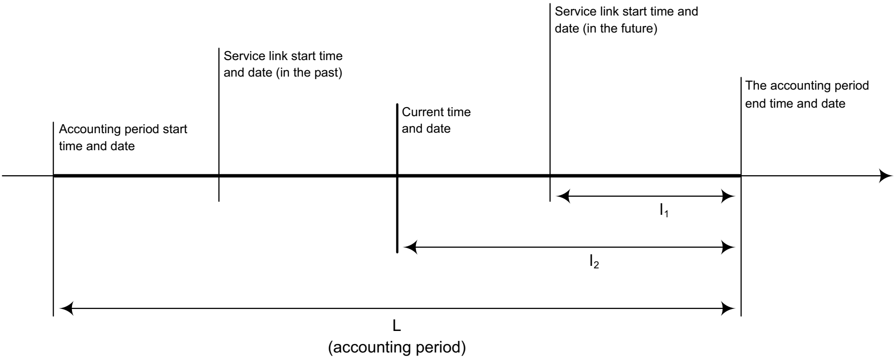
The charge policy allows one to charge the client for the correct amount of money according to the part of the accounting period that he actually was using or is going to use the service.

The cost of a service is not only influenced by the cost that is set when creating it, but is influenced by the charge policy.

Along with the cost of services the charge policy allows to correct the service parameters such as the amount of prepaid traffic or the amount of free minutes (for dial-up service).

Periodic cost component recalculating rules

UTM5 uses the following rules for recalculating the periodic part of the service cost:


Service link start time and date (in the future)
Service link start time and date (in the past)
The accounting period end time and date
Current time and date
Accounting period start time and date
l_1
l_2
L
(accounting period)

The time and date of the service link creation may not match the current time and date and may be set in the future or in the past. If the date and time of the service link creation is set in the past, the current date and time is used instead.

This means that:

Recalculated price = (Full price for accounting period) × `l1/L`,

if the starting date has been set in the future, or otherwise

Recalculated price = (Full price for accounting period) × `l2/L.`

The same rules are used to recalculate the amount of prepaid traffic or prepaid calls duration.

Recalculation when blocking

The current UTM5 version has three types of blocking - administrator’s, system and user’s block.

- **Administrator’s block** is triggered by the administrator when the user’s account needs to be blocked manually.
- **User’s block** is triggered by the user when he doesn’t plan to use the service for some time (e.g. going on vacation).
- **System block** is triggered automatically when the user’s account balance becomes negative. Or, in a certain setup, when the user’s account doesn’t have enough cash balance for charging it with the periodic fee.

For each blocking type the charge policy allows one to set up the following parameters:

- **do not charge periodic fee** when the account is blocked
- **recalc periodic fee** when the account is blocked
- **decrease prepaid traffic** when the account is blocked
- **recalc prepaid telephony** when the account is blocked

When recalculating the amount of prepaid traffic, prepaid calls or the periodic fee, it decreases corresponding to the part of the accounting period during which the account was blocked.

Repay

One may need a repayment i.e. when a user's account was charged with a periodic fee for the whole accounting period and during the period user decides to trigger a voluntary blocking.

The charge policy allows one to set up when exactly does the repayment take place:

- On block expire
- On payment
- On charge period end
- On remove service link

System blocking settings

The charge policy settings also include the settings for the system block.

If an account's balance is not enough to charge it with the periodic fee for the next accounting period, the system block will occur. The charging policy allows one to set up the system blocking to trigger before or after charging user's account. This can be set up separately for each user's service link.

The periodic fee may be adjusted in the service link properties.

Periodic fee charges are done for several services is done in an arbitrary order. If personal account is blocked after a charge for one of the services, charges for the remaining services are done according to the charge policy for accounts in system blocking.

Time ranges

A *time range* is a set of periods of time. Time ranges are used for setting up dependency of a service cost on time and date. For instance, to arrange lower tariffs at night time, one has to create a time range with time limits of 2:00 a.m. till 8:00 a.m., Sunday till Saturday, then create a separate traffic class and relate the new time range to it, and finally to create a service including this traffic class and assign this service to the users.

Interface for working with time ranges is described in **Administrator's interface: Time ranges** on page **70**.

Currencies

The UTM5 system may work with any number of currencies. All personal accounts and charges are made in internal conventional units, so the actual currencies are only used for payment processing and billing. When a payment is processed, the payment currency is converted to internal units. On the other hand, when a bill is issued, the internal units are converted to some currency.

A currency is characterized by its identifier, short name, full name, and the exchange rate (regarding standard unit and discretionary interest coefficient, which multiplies the official rate to get the provider's internal rate). The history of exchange rate since the system's deployment is also available in order to perform financial operations *post factum*.

Interface for handling system currencies is described in **Administrator's interface: Currency** on page **72**.

Each user is associated with some preferred currency for billing. By default, it is defined by the `system_currency` system parameter (see **Administrator's interface: Parameters** on page **82**). The preferred currency may be changed at any moment. As a result, all bills will use the newly selected currency, no matter where they created before or after its selection.

See **User: Other** on page **46** for setting the user's preferred currency.

Payments

There are several ways to make a *payment*, including:

- automatic payment via e-payment systems;
- automatic payment via any third-party software using the `utm5_payment_tool` utility;
- manual payment by an administrator, dealer, or cashier via the UTM Control Center.

Manual payment is made by an administrator or other operator via the Payment page of the administrator's interface, which may be called by the **New payment** from the list of users or from the user details window. In the payment dialog the operator enters the sum of payment, currency, payment date, and probably some other data. In particular, one of the optional parameters is the number of internal or external billing document which is the reason for the payment.

There is an option to define whether the Internet should be switched on for the account, in case if the account balance after the payment allows that. If not checked, then Internet status would not change on payment.

A payment may be provided with the expiration date. These are called *expiring payments* and summed up in a separate report. If an expiring payment has not been spent till its expiration date, i.e. if the sum of charges since the payment date is less than the payment sum, then the rest of the sum is expired (gets withdrawn from the client's account). However, if more expiring payments come in before the expiration date, the expiration of all these payments is postponed till the latest of their expiration dates.

There is a special payment method called *credit*. Such payments are displayed in the user account balance under special category and are obliged to have the expiration date, on which they are undone. When undoing a credit payment, the total credit of the given account is checked, and if it is about to turn negative after withdrawal, the sum of withdrawal is decreased so as to adjust the total credit to zero. Total credit of an account may also be set manually to arbitrary value by the administrator (see **User account** on page **107**).

The interface of making payments is described in **Administrator's interface: Payment page** on page **106**.

Payment rollback

UTM5 billing system has an option of payment rollback. Payments can be rolled back by an administrator or an operator via the UTM Control Center. Nominally, the rollback is done by making a payment of special method (**Rollback**) having the opposite sum.

The rollback procedure is not applicable to the expiring or credit payments.

Rolling back is done via the context menu in the report on payments, see **Administrator's interface: Report on payments** on page **78**.

Documents

The UTM5 system contains several kinds of *documents*. The documents are generated from templates (see **Settings: Document templates** on page **93**). User's contracts exist in an individual manner, are accessible via the administrator's interface (see **User: Contacts** on page **46**) and may be edited after generation. The rest of documents (info sheets, receipts, invoices, etc.) are generated from the templates immediately prior to use.

Invoices

Invoices for services are summarized in the special report (see **Report on invoices** on page **79**). Invoices may be created either automatically or manually. Manual invoices have no effect on the user's account balance.

By default, the invoices for the periodic services and for the periodic portion of special services' price are issued at the end of an accounting period. If the user's account has its **Payment in advance** option checked (see **User: Main** on page **45**) and the service's charge method is set as **At the beginning of the period**, this behavior is reversed so that the invoices are issued at the beginning of the period.

An invoice for a one-time service is issued immediately upon attachment of the service.

Items in the automatic invoices are aggregated by tariff links (with the exception of telephony services, if any, which remain separated from the rest) and by accounting periods. Charges for the new services (those added during the current period) are also not aggregated and stay in a separate invoice, if **Payment in advance** is checked.

After having been generated from a template, an invoice may be edited for printing, but the changes can not be saved.

Invoices with negative VAT rate are hidden in the report.

IP addresses

Multiple kinds of system objects contain subnet address (i.e. an IP address with a mask) as one of their properties. As a rule, an IP address and a mask are entered in the administrator's interface via one common input field as `<address>/<number of significant bits>`. If no mask is entered, this is interpreted as an extremely narrow subnet consisting of single address.

UTM5 supports both IPv4 and IPv6 address formats. IPv6 addresses should be entered in the standard colon-separated form, with possible omission of consecutive zero sections. For example, 2001:db8::ae21:ad12 is the equivalent of 2001:0db8:0000:0000: 0000:0000:ae21:ad12.

ADMINISTRATOR’S INTERFACE

6

Introduction

UTM control center is a program used to control user accounts and billing system settings. Examples of use are given in **Usage examples** on page **129**.

Interface version must be the same as the UTM5 core version, otherwise the interface may work incorrectly.

Top-level menu is described in **Menu**. General interface features and common principles are described in **Common features** (page **43**). The basic interface pages are accessible via the links in the left pane which are grouped in the following sections:

- **Users and groups** (page **44**);
- **Messages** (page **53**);
- **Tariffication** (page **54**);
- **Reference book** (page **71**);
- **Reports** (page **73**);
- **Settings** (page **82**);
- **Interfaces** (page **98**);
- **Additional Features** (page **102**);
- **About** (page **105**);

Depending on the settings and permissions of the particular operator, some pages may be hidden and inaccessible. If all pages in some group are hidden, the group itself is hidden too.

The pages which are accessible neither directly from the left pane nor from the basic interface pages with standard **Add** / **Edit** buttons are described in **Stray pages** (page **106**).

Once started, the program is represented by an icon in the system tray (see **Tray icon** on page **116**).

Menu

Below is the description of the program’s top-level menu.

System

- **Reconnect –** stops the control center and opens the connection window (see **Usage examples: Installation and startup** on page **129**).

- **Import –** opens the **Import** window (see **Structured data import** on page **213**).
- **Exit –** closes the control center.

Settings

- **Properties –** opens the **Properties** window containing the following system settings:
    - **UsersPerPage –** is the number of users per one page as shown in the list of users (see **Users** on page **44**);
    - **CSVSeparator –** is the separator symbol for the exported CSV files (either a comma or a semicolon);
    - **TurnInternetOn –** is the default setting of **Internet status** switch for newly created accounts;
    - **NotVPN –** is the default setting of **Not VPN IP group** flag for newly created IP groups;
    - **DoNotAffectFW –** is the default setting of Do not affect firewall rules flag for newly created IP groups;
    - **UseCustomPassword –** is the flag to use custom set of symbols to generate passwords for newly created users;
    - **CustomPasswordCharset –** is the set of characters used to generate passwords for new users, once the **UseCustomPassword** option is checked;
    - **CustomPasswordLength –** is the generated passwords’ length, once the **UseCustomPassword** option is checked;
    - **DoubleRounding –** is the number of digits after the decimal point to which all the output sums are rounded;
    - **FirstTime –** is the flag of running the program for the first time;
    - **Language –** is the interface language;
    - **SavePassword –** is the flag that controls whether or not to save the password;
    - **SaveSettings –** is the flag that controls whether or not to save the settings.
- **Shortcuts –** opens the **Shortcuts** window for setting hot keys to various typical activities. This window contains the following tabs:
    - **Main window –** contains shortcuts for the pages available in the side menu of the main window of the administrator’s interface;
    - **User –** contains shortcuts for the pages available in the side menu of the edit user window;
    - **Dealer –** contains shortcuts for the pages available in the side menu of the main window of the dealer’s interface;
- **Time zone –** selects the time zone from the drop-down list.

Normally, time zone is imported from the OS settings while installing UTM5.

- **Messages –** sets up the message preview options (see **Messages** on page **53**).

Help

- **About –** displays the version info.

- **Contents –** opens NetUP UTM5 help.

Common features

The basic interface pages contain lists of entities (users, services, etc.) with a number of features for each entity.

Mouse right-clicking on the list reveals a context menu that contains a number of handy commands including **Edit**, **Remove** (once the operator has sufficient privileges), **Refresh**, **Columns**, and probably also some commands specific to a particular type of entities.

Columns of the tables can be reordered by drag-and-dropping. Their width may also be adjusted by dragging the column borders. The representation of particular columns may be switched on or off via the context menu item named **Columns**. As a rule, the default settings imply that all possible columns are displayed.

Entries in the list may be ordered by any column via clicking on the column header. Repeated clicking on the same column header reverses the sort order.

Multiple selection of list items may be performed by left-clicking with pressed **Shift** (selects a range of entries) or **Ctrl** (selects multiple entries one by one). Pressing **Ctrl** + **A** selects all entries on the page.

The **Export** item of the context menu exports the list in the CSV or XML format, considering the current column display settings.

Depending on the purpose of the page and the user's rights, the page may contain an interface for addition, editing or removal of its elements in a form of buttons **Add**, **Edit**, and **Delete**. Some of the buttons may be disabled due to the insufficient access rights. In case of view-only access the **Edit** button is substituted by **Read**. As a rule, the addition or editing of an element is performed in a separate window. The corresponding form may or may not have the Reset button which resets all its fields at once.

The Refresh button refreshes the list to reflect possible changes that could have been introduced by another administrator in the meantime, or could have occurred automatically.

The top menu (see **Menu** on page **41**), the left pane with quick links, and the bottom status line displaying current server time are visible and accessible from any interface page.

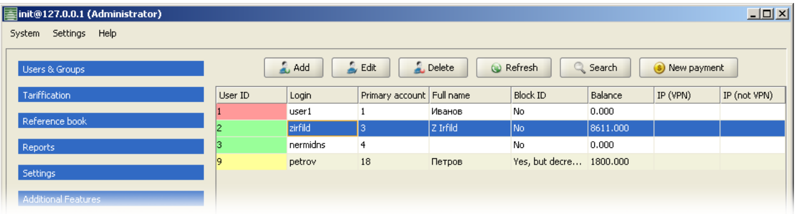
In case if the connection to UTM5 core is lost, focus switches from the main window to the pop-up window with **Reconnect** button.

Users and groups

Users

This page contains the list of users (see **Basic system objects: Users** on page **26**) with the interface for creating, removing, editing a user, or making payment. The list contains the following info about each user:

- **User ID –** is the ID of the user in the system.
- **Login –** is the user’s login.
- **Primary account –** is the account number.
- **Full name –** is the full name of the user or a title of the legal entity.
- **Block ID –** is the blocking status of the user.
- **Balance –** is the account balance.
- **IP (VPN) and IP (non-VPN) –** are the lists of user’s networks set in the properties of IP traffic service links.



User ID	Login	Primary account	Full name	Block ID	Balance	IP (VPN)	IP (not VPN)
1	user1	1	Иванов	No	0.000		
2	zirfild	3	Z Irfild	No	8611.000		
3	nermidns	4		No	0.000		
9	petrov	18	Петров	Yes, but decre…	1800.000		

Page contains the following interface elements:

Add button opens the user creation window (see **Adding users** on page **45**).

Edit buttons open the user details window that includes a number of interface pages accessible via the quick links on the left pane, which are gathered into the following groups: **User** (page **45**), **Tariffication** (page **46**), and **Reports** (page **47**).

Delete button removes the selected user(s), once the related service links and tariff links are removed, or displays an error message otherwise.

Search button opens the search window (see **Search page** on page **107**).

New payment button opens the payment window (see **Payment page** on page **106**).

Context menu of the list of users contains quick links for the following operations with the selected user(s):

- Switch Internet on;
- Switch Internet off;

- Make a payment.

Unlike the majority of other lists, the list of users is displayed pagewise, with the page number and the number of users per page set in the bottom part of the page. These settings are persistent, i.e. once set, they are saved and resumed on the next launch of the program.

List entries are marked with color. Red means that the user’s accounts are blocked, green means they are not, and yellow means that some accounts are blocked, while others are not.

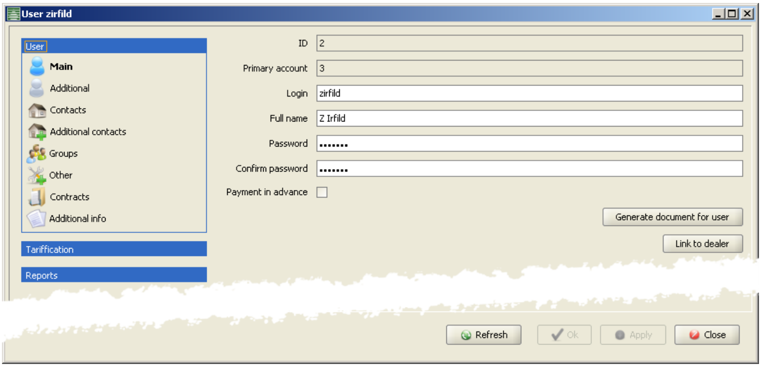
Adding users

The **Add user** window contains the following pages:

- **Main –** includes login, full name, password, and the **Payment in advance** check box. The login is checked for uniqueness, and may include the following symbols: lowercase letters (from a to z), numerals (0-9), dot, comma, at mark (@), underscore (_), hyphen (-), and slash (/). You may choose to generate a random password, which is then automatically substituted into the **Password** and **Confirm password** fields, and shown openly in the **Generated password** field for copying.
  This login:password pair is used solely for the access to the user interfaces (see **UTM5 tray utility** on page **233** and **Web interface** on page **239**).
- **Additional –** includes bank account details and some other data, including custom parameters (see **Settings: Additional parameters**).
- **Contacts –** includes user’s personal data (address, phone, e-mail).
- **Other –** are special parameters associated with the user, including remote switch address, port, and preferred currency.

User

The **User** group in the user properties window includes the following pages:



ID: 2
Primary account: 3
Login: zirfild
Full name: Z Irfild
Password: •••••••
Confirm password: •••••••
Payment in advance: ☐

- **Main –** includes login, full name, password, and the following elements:

  - **Payment in advance –** check box (if checked, the invoices for the periodic services with charge method set as **At the beginning of the period** are issued at the beginning of the accounting period; has no effect on charges);
  - **Generate document for user –** button that displays the handout document for the user containing login, password, and the provider’s contact information;
  - **Link to dealer –** button that opens the interface to link this user to one of the existing dealers (see **Dealer module** on page **221**).
- **Additional –** includes bank account details and some other data, including custom parameters (see **Settings: Additional parameters**). Bank details may be filled in automatically by linking the user to a bank (see **Banks** on page **73**).
- **Contacts –** includes personal data (address, phone, e-mail) of the contact person. The address may be filled in automatically by linking the user to a house (see **Buildings** on page **72**). Also, this page contains the **Send invoices by email** check box that gets active once an e-mail is entered.
- **Additional contacts –** includes personal data of additional contact persons, if any.
- **Groups –** is the list of groups the user belongs to, together with the interface to add the user to a group or remove from it.
- **Other –** are special parameters associated with the user, including document profile, remote switch address, port, and preferred currency. For more information about document profiles see **Document profiles** on page **95**.
- **Documents –** is the list of documents for the user, together with the interface to generate, edit and delete them. Documents may be generated from templates (see **Document templates** on page **93**), or may be uploaded as *.odt files.

*Any *.odt file may be uploaded as user document.*

- **Additional info –** is the view-only auxiliary information (dates of creation and last modification of the user).

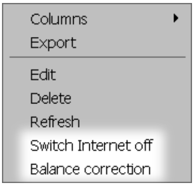
Tariffication

The **Tariffication** group in the user properties window includes the following pages:

- **Accounts –** is the list of the user’s accounts containing the interface to create, edit, and remove the accounts, as well as to make payments. Creation and editing of an account is performed in a special window (see **User account** on page **107**).

  For usage examples see **Creating account** on page **134** and **Removing an account** on page **135**.

To change the blocking settings of an already blocked account, it is necessary to lift the existing blocking and then impose it again with new settings.



Right click one of the listed accounts to open the context menu. This menu, along with standard elements, contains the following:

  - **Switch Internet off –** switches off Internet for the selected account
  - **Balance correction –** allows one to correct balance for the selected account. One can also add a comment for this action. Balance correction operations and comments for these operations get to the *User Change Log* report.

- **Service links –** is the list of the user's service links together with the interface to create, edit, and remove them, also containing the following interface elements:
  - **Prepaid traffic –** sets the prepaid traffic (active if the item selected in the list is an IP traffic service link);
  - **Set RADIUS parameters –** sets the RADIUS attributes for the selected service link;
  - **Select account –** is the drop-down list to select one of the user's accounts.

  Creation and editing of a service link is performed in a special window (see **Service link** on page **109**).

  For usage example see **Creating service links** on page **136**.
- **Tariff links –** is the list of the user's tariff links together with the interface to create, edit, and remove them, also containing the following interface elements:
  - **History –** displays the history of tariff plans previously associated with the user;
  - **Select account –** is the drop-down list to select one of the user's accounts.

  Creation and editing of a service link is performed in a special window (see **Tariff link** on page **108**).

  For usage example see **Creating tariff links** on page **138**.
- **Technical parameters –** are the arbitrary parameters associated with the user. Their values may be used in the commands for controlling the external software, which are sent by UTM5 as a response to certain events, see **UTM5 RFW: Firewall rules** on page **186**.
- **IPTV activation codes –** is a list of activation codes for an access card, assigned to the selected personal account.

Reports

The **Reports** group in the user properties window includes the reports of various types for the selected user, each on a separate page. The interface is similar to that on the general reports page (see **Reports** on page **73**), except for the following details:

- Group selector is missing.
- On all pages, except for **Detailed Traffic**, **Graphic Report** and **User Change Log**, there is a drop-down list to select one of the user's accounts, or all accounts.
- On the **Invoices** page (see below) there is a **New invoice** button.

  The following types of reports are included:

- **General –** is the report on payments and charge-offs of all types;

- **Blockings –** is the report on blockings of the given user;
- **Traffic –** covers the consumption of traffic by classes;
- **Telephony –** contains the statistics of phone calls;
- **Telephony directions –** contains aggregated statistics of phone calls by directions;
- **Sessions –** contains the statistics of dialup and VPN sessions;
- **Payments –** contains the statistics of payments to the user’s account(s);
- **Services –** contains the statistics of services;
- **Other charges –** lists the charges not related to any services (payment expiration, rollback, etc.);
- **Internal transfer –** lists the transfers of funds between one’s accounts, which the users may perform on their own;
- **Detailed Traffic –** covers the traffic consumption in full detail, with source and destination addresses and port numbers;
- **Invoices –** contains the statistics of invoices. Also contains the **New invoice** button, which allows for creation of an invoice with arbitrary positions;
- **User Change Log –** covers the changes in the user’s data;
- **Expiring Payments –** contains the statistics of expiring payments;
- **Graphic Report –** contains the reports on some services (IP traffic, dialup, telephony) in a graphic form;
- **Custom charges report –** lists the charges performed by third-party systems via integration modules.
- **DHCP lease –** lists the history of DHCP leases in selected time range.

Card users

This page contains the list of card users, which are the users generated automatically on registration of prepaid cards on the automatic registration page of the web interface (see **Web interface: Entrance page** on page **240**). Card users are not supposed to be created manually.

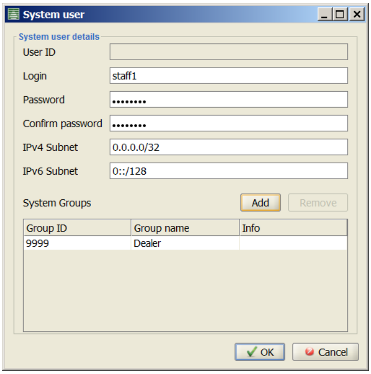
Page contains the following interface elements:

- Edit buttons open the user details window similar to that of a regular user, i.e. containing the following page groups: **User** (page **45**), **Tariffication** (page **46**), and **Reports** (page **47**).
- Delete button removes the selected card user(s), once the related service links and tariff links are removed, or displays an error message otherwise.
- New payment button opens the payment window (see **Payment page** on page **106**).
- Clear button removes the card users linked to overdue cards.

The interface of card generation is described in **Card pools** on page **51**.

System users

This page contains the list of system users (see **Basic system objects: System groups** on page **28**) with the interface for creating, removing, or editing them.



User ID	
Login	staff1
Password	••••••••
Confirm password	••••••••
IPv4 Subnet	0.0.0.0/32
IPv6 Subnet	0::/128

System Groups

Group ID	Group name	Info
9999	Dealer	

The Add and Edit buttons open the system user properties window containing the following input fields:

- **User ID –** of the user.
- **Login –** of the user in the system.

Logins of the system users can not coincide with those of the ordinary users.

- **Password, confirm password –** are the password and its confirmation fields.
- **IPv4/IPv6 subnet –** is the subnet from which the user’s access is allowed (optional parameter). See **IP addresses** on page **40** for the formatting details.

Besides that, a system user may be included in one or more system groups (see **System groups** on page **50**) providing access to a certain set of functions.

Any changes to the list of groups take effect only after the next UTM5 core restart.

Dealers

This page contains the list of dealers with the interface for creating, removing, or editing them. Dealers are the special entities that may perform some administrative functions over a particular subset of users.

See **Dealer module** on page **221** for the details on dealers’ purpose, functions, creation, and operation.

Groups

This page contains the list of user groups with interface to create, edit, and remove them, as well as to perform group operations.

Editing of a group may include removal of users from the said group. Addition of users to a group may be performed on the user properties page (see **User: Groups** on page **46**) and on the search page (see **Search page** on page **107**).

Group operations include:

- **Switch Internet on –** for all members of the selected group;

- **Switch Internet off –** for all members of the selected group;
- **Group all blocked users –** (i.e. all blocked users are added to the selected group);
- **Set tariff plan for next accounting period –** for the members of the selected group having particular current value of the next tariff plan, or for the whole group (if the value is specified as “Any”).

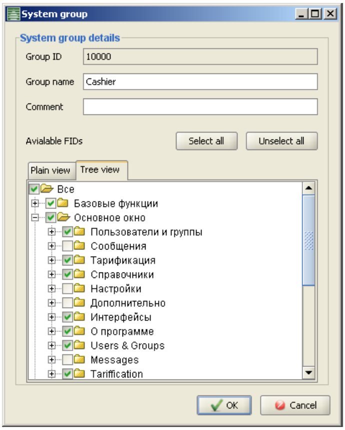
Each user is normally associated with some current tariff plan and some tariff plan for the next accounting period. These two plans may or may not coincide. The selection is based upon the latter.

- **Change policy –** for the members of the selected group (see **Charge policy** on page **35**).

System groups

This page contains the list of system groups with the interface for creating, removing, editing, or copying them. System groups are used to set the permissions of system users (see **System users**).

The Add and Edit buttons open the **System group** window containing two tabs, namely **Plain view** and **Tree view**.



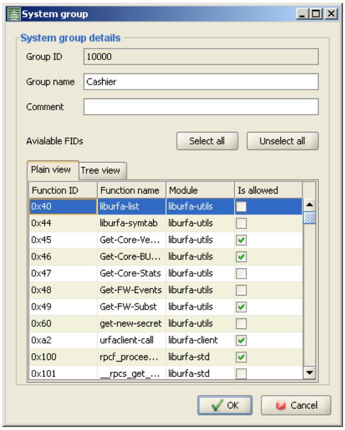
Group ID 10000

Group name Cashier

Comment

Avialable FIDs

- ☑ Все
  - ☑ Базовые функции
  - ☑ Основное окно
    - ☑ Пользователи и группы
    - ☐ Сообщения
    - ☑ Тарификация
    - ☑ Справочники
    - ☐ Настройки
    - ☐ Дополнительно
    - ☑ Интерфейсы
    - ☑ О программе
    - ☑ Users & Groups
    - ☐ Messages
    - ☑ Tariffication



Group ID 10000

Group name Cashier

Comment

Avialable FIDs

Function ID	Function name	Module	Is allowed
0x40	liburfa-list	liburfa-utils	☐
0x44	liburfa-symtab	liburfa-utils	☐
0x45	Get-Core-Ve…	liburfa-utils	☑
0x46	Get-Core-BU…	liburfa-utils	☑
0x47	Get-Core-Stats	liburfa-utils	☐
0x48	Get-FW-Events	liburfa-utils	☐
0x49	Get-FW-Subst	liburfa-utils	☑
0x60	get-new-secret	liburfa-utils	☐
0xa2	urfaclient-call	liburfa-client	☑
0x100	rpcf_procee…	liburfa-std	☑
0x101	__rpcs_get_…	liburfa-std	☐

Each system group has the following parameters: ID, name, comment, and the list of available functions. The latter may be picked via any of the two interfaces:

- On the **Tree view** tab the visual interface is presented, with all functions grouped in the hierarchical tree by application field. The tree is redundant, so that some low-level functions are included in multiple branches simultaneously. On changing the permission for such a function a warning window appears with a listing of other branches affected by this permission.

- On the **Plain view** tab all functions are displayed in an unstructured alphabetic list.

The ***Wheel*** *and* ***Dealers*** *system groups are built-in, so their properties can not be altered. It is possible, though, to create copies of these system groups (note the* ***Copy*** *item in the context menu) and fiddle with their properties to match your needs.*

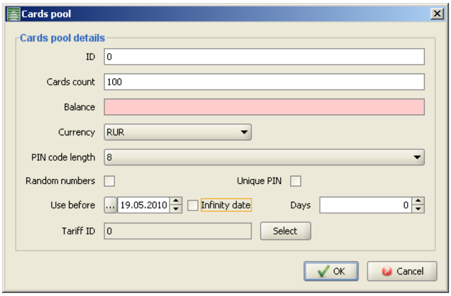
Card pools

This page contains the list of prepaid cards pools (see **Basic system objects: Prepaid cards** on page **28**), with the interface to add or modify them. Card pools can not be removed. Page contains the following interface elements:

- Add button opens the card generation window (see **Generating cards** on page **51**).
- Edit button opens the card pool properties window (see **Editing card pools** on page **52**).
- Clear button removes overdue cards from the selected pool(s).
- Search button opens the cards search window, where the cards may be searched by an arbitrary combination of conditions on card ID, pool ID, tariff ID, PIN code, card balance, currency, and the activation date.

Generating cards

Card generation window contains the following fields:



Cards pool

Cards pool details

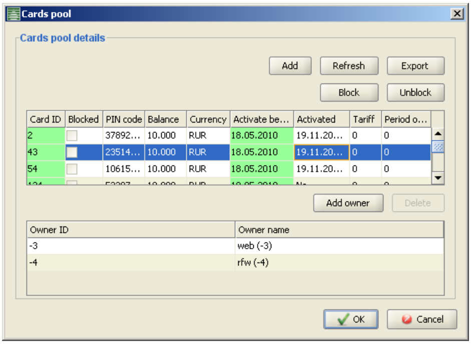
ID: 0
Cards count: 100
Balance:
Currency: RUR
PIN code length: 8
Random numbers ☐
Unique PIN ☐
Use before: 19.05.2010 ☐ Infinity date
Days: 0
Tariff ID: 0

- **ID –** is the pool ID (if points at the existing pool, the cards are added to it; otherwise, a new pool is created).
- **Cards count –** is the number of cards to be generated.
- **Balance –** is the monetary value of one card.
- **Currency –** is the currency in which the card value is specified.
- **PIN code length –** is the number of digits in the PIN codes to be generated.
- **Random numbers –** switches on generation of random ID numbers for the cards; if not set, the numbers are issued sequentially.
- **Unique PIN –** requires that the generated PIN codes are unique.
- **Use before –** is the optional date to activate the card strictly before.
- **Days –** is the optional term of expiration of the payment made when the card gets activated. If not set, the payment is not expiring.
- **Tariff ID –** is the optional tariff attached to the card users on registration.

The created cards may be neither edited nor removed.

If the tariff for the card users contains some services with periodic component, on user’s registration those get attached to the system accounting period, which is since 1/1/1970 till 1/19/2038. Therefore only the services with no or negligible periodic payment are suitable to include in such tariff plans.

Editing card pools



Card ID	Blocked	PIN code	Balance	Currency	Activate be...	Activated	Tariff	Period o...
2	☐	37892...	10.000	RUR	18.05.2010	19.11.20...	0	0
43	☐	23514...	10.000	RUR	18.05.2010	19.11.20...	0	0
54	☐	10615...	10.000	RUR	18.05.2010	19.11.20...	0	0

Owner ID	Owner name
-3	web (-3)
-4	rfw (-4)

Card pool details window lists the cards in the pool together with their PIN codes, status info, and activation dates. The window contains the following interface elements:

- **Add –** button opens the **Generating cards** window to add a new lot of cards to the pool.
- **Refresh –** button updates the list of cards.
- **Export –** button exports card data for this pool to an XML file.
- **Block –** button blocks selected cards.
- **Unblock –** button unblocks selected cards.

Below is the list of owners. These are the system users having the right to register users based on the cards from this pool. The option of having different owners for various pools may be relevant when the system contains several web interfaces run by different system users. If the list is not set, any system user has the right to register users.

The list may be altered by the following buttons:

- **Add owner –** add a new owner to the list;
- **Delete –** deletes the selected owner from the list.

IP groups

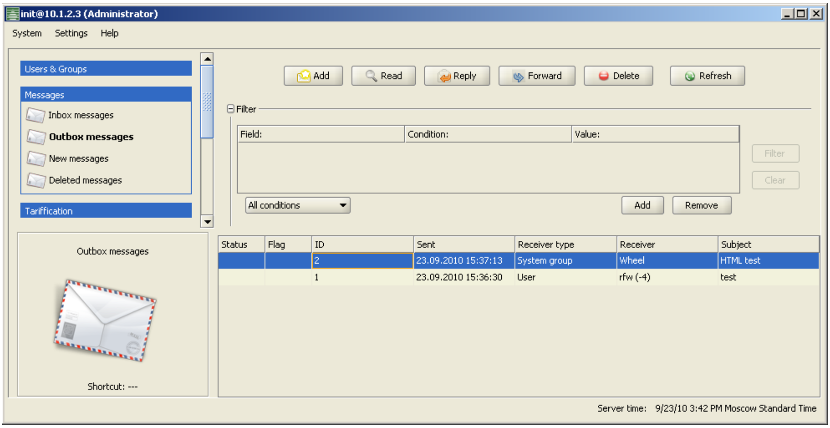
This page contains the list of IP groups defined within service links of IP traffic services (see **IP traffic service link** on page **111**).

Telephone numbers

This page contains the list of telephone numbers defined within telephony service links (see **Telephony service link** on page **115**).

Messages

This page contains interface for sending and receiving system messages from the users and other administrators. Users may send their messages via web interface (see **Web interface: Messages** on page **242**) or via the `utm5_tray` application (see **UTM5 tray utility: Messages** on page **236**). If the `web_message_group` parameter (see **Interface parameters** on page **152**) is set, then the messages sent via web interface will be seen only by the members of the system group specified by this parameter.



Status	Flag	ID	Sent	Receiver type	Receiver	Subject
		2	23.09.2010 15:37:13	System group	Wheel	HTML test
		1	23.09.2010 15:36:30	User	rfw (-4)	test

Server time: 9/23/10 3:42 PM Moscow Standard Time

Messages are subdivided into the folders:

- Incoming;
- Outgoing;
- New;
- Deleted.

The **Filter** roll-up pane may be used to filter messages by an arbitrary set of conditions on the message parameters, which include sender, receiver, sending time, etc.

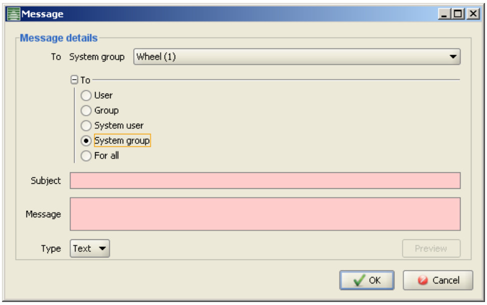
Depending on the system settings (see **Settings: Messages** on page **42**), the preview pane for viewing the selected message may be shown along with the list of messages.

The following action buttons are present:

- **Add –** creates a new message.
- **Read –** opens the selected message in a separate window for reading.
- **Reply –** creates a new message in a reply to the selected one.
- **Forward –** resends the selected message to another addressee.

- **Delete –** moves the selected message to the **Deleted** folder.

A message may be addressed to:


Message

Message details

To System group: Wheel (1)

To
○ User
○ Group
○ System user
● System group
○ For all

Subject

Message

Type: Text

- User (selected in a separate search window, see **Search page** on page **107**);
- Group (selected from the list);
- System user (selected from the list);
- System group (selected from the list);
- All users.

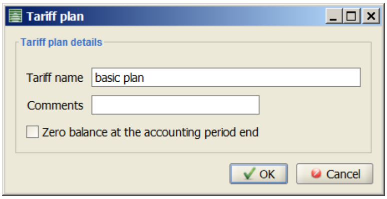
Message type may be specified as either **Text** or **HTML**.

Tariffication

Tariff plans

This page contains the list of registered tariff plans (see **Basic system objects: Tariff plans** on page **30**) with the interface for creating, removing, or editing them.

A tariff plan can be removed only if it is not used at the moment. Otherwise, in the first place it is necessary to remove all tariff links attached to it.


Tariff plan

Tariff plan details

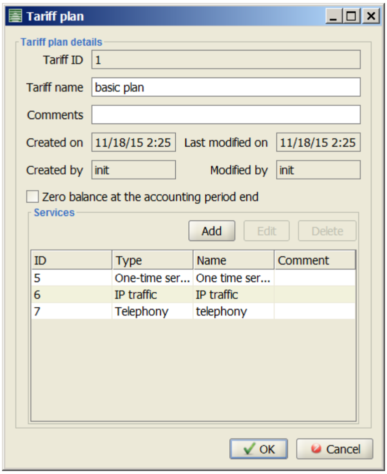
Tariff name: basic plan

Comments:

☐ Zero balance at the accounting period end

Add button opens the tariff plan creation window with the following input fields:

- **Tariff name –** is a mandatory parameter.
- **Zero balance at the end of accounting period –** is a check box for resetting to zero the balance of the account connected to this tariff plan by the end of the accounting period.



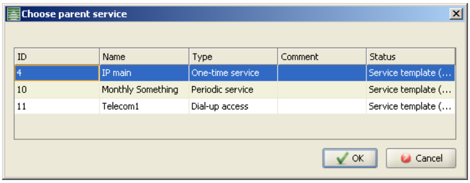
Tariff ID: 1
Tariff name: basic plan
Comments:
Created on: 11/18/15 2:25 Last modified on: 11/18/15 2:25
Created by: init Modified by: init
☐ Zero balance at the accounting period end

ID	Type	Name	Comment
5	One-time ser...	One time ser...	
6	IP traffic	IP traffic	
7	Telephony	telephony	

Edit button opens the tariff plan properties window that contains the following parameters:

- **Tariff ID –** is assigned automatically.
- **Tariff name –** is a mandatory parameter.
- **Created on, Last modified on –** are the dates of creation and the last modification of the tariff plan.
- **Created by, Modified by –** are the names of the system users responsible for the tariff plan creation and its last modification, correspondingly.
- **Zero balance at the end of accounting period –** is a check box for resetting to zero the balance of the account connected to this tariff plan by the end of accounting period.
- **Services –** is the list of services included in the tariff plan.

To add a new service to the tariff plan, press **Add**. The list of existing service templates will show up.



ID	Name	Type	Comment	Status
4	IP main	One-time service		Service template (...
10	Monthly Something	Periodic service		Service template (...
11	Telecom1	Dial-up access		Service template (...

Select the template you need and press **OK**.

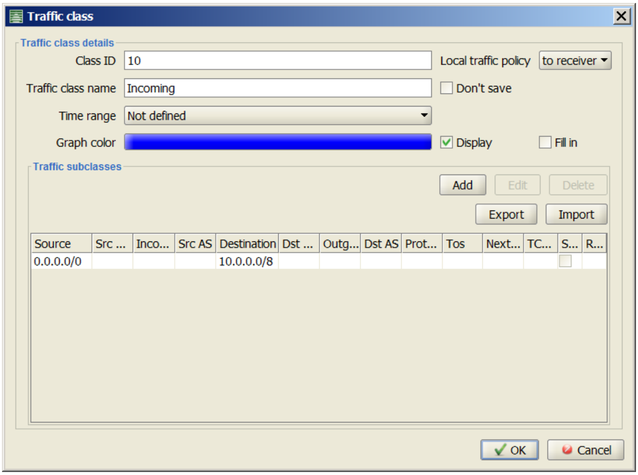
A tariff plan may not contain multiple services originating in the same template.

Traffic classes

This page contains the list of registered traffic classes (see **Basic system objects: Traffic classes** on page **28**) with the interface for creating, removing, or editing them. A traffic class can be removed only if it is not used at the moment. Otherwise, in the first place it is necessary to remove all entities (services, etc.) that refer to it. Once a traffic class is removed, the traffic belonging to it would pass for unidentified in the control center reports (see **Administrator’s interface: Reports** on page **73**). At the same time, in the web interface reports (see **Web interface: Reports** on page **242**) the removed traffic class would continue to show up as existing.

For the example of usage see **Creating traffic classes** on page **131**.

Below is the list of traffic class parameters and their meanings:



Class ID: 10

Traffic class name: Incoming

Time range: Not defined

Graph color

Local traffic policy: to receiver

☐ Don't save

☑ Display

☐ Fill in

Source	Src ...	Inco...	Src AS	Destination	Dst ...	Outg...	Dst AS	Prot...	Tos	Next...	TC...	S...	R...
0.0.0.0/0				10.0.0.0/8								☐	

- **Class ID –** is a mandatory parameter. The numbers should be selected in such a way that the numbers of child classes would be higher than that of their parent class.
- **Traffic class name –** is a mandatory parameter.
- **Don’t save –** check box disables saving of raw traffic data files, if checked. May be worthwhile to set for the free traffic or in other cases when the detailed information is unlikely to ever become necessary.

Format of the raw traffic data files has been altered in UTM5.3-001, so that the files saved in earlier UTM5 versions can no longer be read.

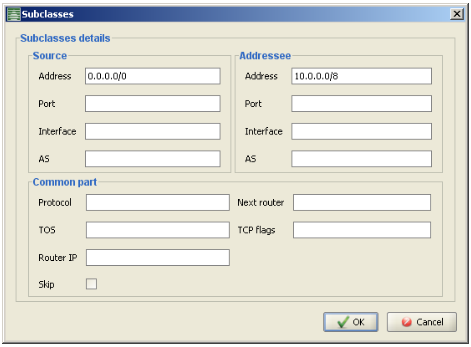
- **Time range –** limits the traffic class existence with the given time range, if selected.
- **Graph color –** is the color to represent the traffic of this class in the graphic reports.
- **Display –** enables display of this class in the graphic reports.
- **Fill in –** sets fill style for the graph reports.

Besides that, the traffic classes page contains interface for working with subclasses. The interface consists of the following buttons.

- **Add –** a new subclass;
- **Edit –** the selected subclass;
- **Remove –** the selected subclass;
- **Export –** all subclasses of the given class as a CSV or XML file with following contents.
    - For **CSV**: each subclass is described on a single line. Properties’ values are separated with semicolons, and each line ends with a semicolon. Names of the properties are listed in the first line.

  - For **XML**: the root tag is `UTM_export`. Each subclass is described in a separate instance of the `row` tag. Inside it there is the tag `row_id` followed by the subclass properties, each wrapped in its own dedicated tag, in the same order as in the CSV file.
- **Import –** imports a CSV or XML file of similar format and attaches all retrieved subclasses to the given class.

Traffic subclasses



Source: Address 0.0.0.0/0

Addressee: Address 10.0.0.0/8

Traffic subclass is characterized by the following parameters:

- For the source or destination:
  - Network address and mask (mandatory, may be entered as IPv4 or IPv6, see **IP addresses** on page **40** for the formatting details);
  - Port;
  - Interface;
  - AS (autonomous system number);
- Protocol;
- Next router;
- TOS (TCP/IP “type of service” field);
- TCP flags;
- Router IP.

Types (IPv4 vs. IPv6) of the source and destination addresses must coincide; type of the router IP does not have to follow them and may be arbitrary.

If **Router IP** is not set, the NetFlow provider address is not considered.

If **Skip** is checked, the affiliation with subclass is interpreted negatively in the classification, i.e. it means that the traffic does not belong to the given class and needs to be checked against other classes. This may be useful if some address or a group of addresses needs to be pick out as a separate class.

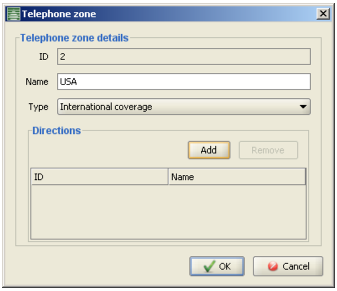
Telephone zones

This page contains the list of registered telephone zones with the interface for creating, removing, or editing them.

Telephone zone is a set of telephone directions (see **Telephone directions**) joined together for more convenient tariffication of phone calls.

The cost of a telephone call may be set in the settings of the Telephony service for any created zone taking into account time ranges.

Telephone zone is characterized by the following parameters:


Telephone zone

Telephone zone details

ID: 2

Name: USA

Type: International coverage

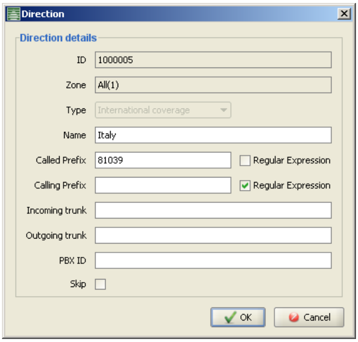
Directions

- ID;
- Name;
- Type of coverage, which is to be selected among:
  - local,
  - inner zone,
  - intercity,
  - international;
- List of telephone directions included in the zone. May be edited by **Add** / **Remove** buttons.

Telephone directions

This page contains the list of registered telephone directions with the interface for creating, removing, or editing them.

Telephone direction is a set of telephone numbers. The attribution of a particular number to a direction is checked by means of regular expressions. Telephone directions are used to classify phone calls for subsequent tariffication.


Direction

Direction details

ID: 1000005

Zone: All(1)

Type: International coverage

Name: Italy

Called Prefix: 81039 ☐ Regular Expression

Calling Prefix: ☑ Regular Expression

Incoming trunk:

Outgoing trunk:

PBX ID:

Skip: ☐

Telephone direction is characterized by the following parameters:

- **ID –** is a number > 1000000 (assigned automatically).
- **Zone –** is the name of a telephone zone into which the direction is included (set automatically).

Every telephone direction may be included in one and only one telephone zone.

- **Type –** is the call type (local, intercity, etc.; inherited from the parent zone, if it is set).
- **Name –** is a mandatory parameter.

  Classification criteria (at least one must be non-empty):

- **Called prefix –** is a prefix or a regexp (POSIX 1003.2 compatible) for checking the called number.
- **Calling prefix –** is a prefix or a regexp for checking the calling number.
- **Incoming trunk**;
- **Outgoing trunk**;
- **PBX ID**;
- **"Skip" flag –** cancels identification (if checked, no calls will be identified into this direction).

The list of directions is kept ordered lexicographically by called prefix, then by calling prefix, then by incoming trunk, and then by outgoing trunk. The search is performed from the beginning of the list till the first match. To be identified with the direction, a call must match all parameters (called prefix, calling prefix, incoming trunk, outgoing trunk, PBX ID) which are set for this direction.

It is recommended to create the default direction with called prefix `^.*$`, so as to leave no number unassigned.

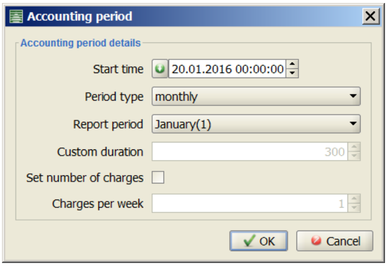
Accounting periods

This page contains the list of current accounting periods (see **Basic system objects: Accounting periods** on page **29**) with the interface for adding or editing them.

An accounting period can not be removed.

Once a period finishes, a new period of the same type is created automatically.

Below is the list of accounting period parameters and their meanings:



Start time 20.01.2016 00:00:00
Period type monthly
Report period January(1)
Custom duration 300
Set number of charges ☐
Charges per week 1

- **Start time –** is the date and time when the period begins.
- **End time –** is the date and time when the period ends. When creating a new period, this field is missing, since the date is calculated automatically.
- **Period type –** is selected among the following:
  - daily;
  - weekly;
  - monthly;
  - quarterly;
  - annual;

  - custom duration.

Monthly period ends in the next calendar month after its start, on the same day of month. However, if the starting date exceeds the number of days in the next month (say, January 30), the period lasts only till the end of the next month.

- **Custom duration –** is the length of period in seconds. Enabled only if **Period type** is set to **Custom duration**. The shortest possible duration is 3600 seconds.
- **Set number of charges –** enables setting number of charges per week.
- **Charges per week –** is the number of periodic charge-offs per week. Enabled only if **Set number of charges** is checked.

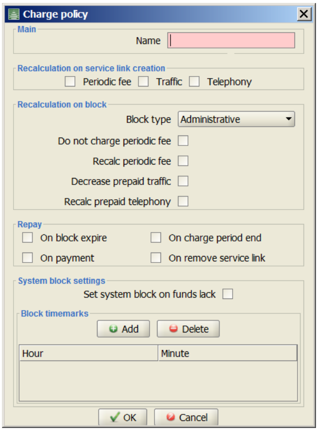
When an existing period is edited, its ending date is the only property that may be changed.

Charge policies

This page contains the list of active charge policies.

Every charge policy contains certain rules for recalculation of periodic component of a service price, prepaid services amount and refunding rules. I.e. it is the periodic fee, prepaid traffic and calls amount recalculation settings.

Creating a charge policy



Charge policy

Main

Name

Recalculation on service link creation

☐ Periodic fee ☐ Traffic ☐ Telephony

Recalculation on block

Block type: Administrative

Do not charge periodic fee ☐

Recalc periodic fee ☐

Decrease prepaid traffic ☐

Recalc prepaid telephony ☐

Repay

☐ On block expire ☐ On charge period end

☐ On payment ☐ On remove service link

System block settings

Set system block on funds lack ☐

Block timemarks

Hour	Minute

Press Add to add a new charge policy. A charge policy has the following parameters:

- **Main**:
  - **Name –** is the name of the charge policy
- **Recalculation on service link creation –** contains parameters that define which of the following will be recalculated on link creation:
  - **Periodic fee –** is the periodic fee
  - **Traffic –** stands for the amount of prepaid traffic
  - **Telephony –** stands for the amount of prepaid calls
- **Recalculation on block –** contains recalculation parameters for different blocking types:
  - **Block type –** is a drop-down menu that allows one to switch to another block type

Switching to another block preserves all the parameters for the previously chosen block type

  - **Do not charge periodic fee –** means that the account won't be charged for the periodic fee while in block

  - **Recalc periodic fee –** means that the periodic fee will be recalculated proportionally to the time spent in block during the current accounting period
  - **Decrease prepaid traffic –** means that the amount of prepaid traffic will be recalculated in the same way as the periodic fee
  - **Recalc prepaid telephony –** means that the amount of prepaid calls will be recalculated in the same way as the periodic fee.

Note that the recalculation parameters are set when blocking starts. This means that when the blocking ends, periodic fee, prepaid traffic and calls will be recalculated according to the charge policy parameters recorded when blocking started, even if the charge policy parameters were changed since then.

- **Repay –** is a set of rules that define which event should be coupled with a customer debt repayment (when the customer's account has been charged excessively):
  - On block expire
  - On payment
  - On charge period end
  - On service link removal
- **System block settings –** has the following parameters:
  - **Set system block on funds lack –** sets system block when an account has insufficient funds for fee withdrawal at the beginning of the next accounting period (fee withdrawal doesn't happen in this case). If this option is disabled, the check is not performed. In that case if an account has insufficient funds, it's balance will become negative after withdrawal and the account will be blocked.
  - **Block timemarks –** is the time for daily check of blocked account's balance. If at some point of time there's enough funds to pay for the services for the rest part of the current accounting period, UTM withdraws that money and unblocks the account. Block timemarks are only used when the first option is enabled.

Services

The **Services** page contains the list of registered services (see **Basic system objects: Services** on page **31**) with the interface for creating, removing, editing, or copying them. The action of copying a service is invoked by the **Copy** item in the context menu.

A service can be removed only if it is not used at the moment. Otherwise, in the first place it is necessary to remove all service links based on it.

Pressing Add opens the service properties window focused on the **Main** page. This page includes the following elements:

- **Name –** is the name of the service.
- **Comment –** is an arbitrary comment.
- **Type –** is the drop-down list for selecting the type of service.

- **Supplier to invoice –** is the legal entity on whose behalf the service is provided (see **Companies** on page **92**).

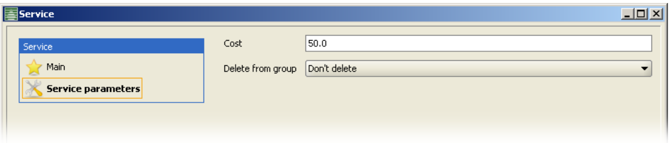
The contents and composition of other pages (see below) varies depending on the selected **Type** value.

On pressing Edit, the service properties window with disabled **Type** field shows up.

Attachment of services to the users is normally done by means of service links created via the user properties page (see **Users: Tariffication** on page **46**).

One-time service

One-time service properties window consists of two pages:



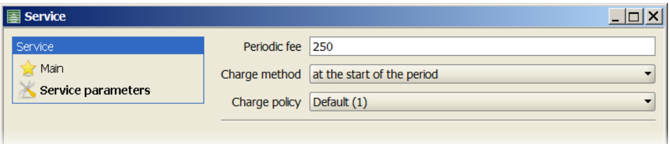
Cost: 50.0
Delete from group: Don't delete

- **Main** (standard, **Above**).
- **Service parameters**. Includes the following elements:
    - **Cost –** is the price of the service.
    - **Delete from group –** is a drop-down list for selecting a group from which the user is to be excluded immediately after the charge-off for the service.

Parameters of the corresponding service link are described in **One-time service link** on page **109**.

Periodic service

Periodic service properties window consists of two pages:



Periodic fee: 250
Charge method: at the start of the period
Charge policy: Default (1)

- **Main** (standard, **Above**).
- **Service parameters**. Includes the following elements:
    - **Periodic fee –** is the fixed price of the service per one accounting period. Services of specialized types may have some other components of price besides this one, namely, the price per unit of traffic or per connection time, etc.
    - **Charge method –** is the order of charging off the user’s account. The possible values are:
        - **At the beginning of the period** means that the charge-off is done at one instant on creation of the service link;

    - **At the end of the period** means that the charge-off is done at once immediately before the closing of the accounting period (the one associated with the service link);
    - **Flow method** means that the charge-off is done in portions during the whole length of the accounting period. The number of portions is determined by the corresponding parameter of the accounting period, if it is set, or by the core settings otherwise (the `flow_discounts_per_period` parameter).
  - **Charge policy –** is the default charge policy that will be used for service link creation for this service. For more information on charge policy see **Basic system objects: Charge policy** on page **35**. See also charge policy creation at **Charge policies** on page **60**.

If the flow method is selected, the charge-off is performed as follows. Based on the price of the service and the length of the accounting period, the minimal one-time payment value is determined. Based on the number of charge-offs, the minimal time between charge-offs is calculated. Once the service link is initialized, the corresponding accounting period is divided into equal parts. At the end of each part, the total cost of the service to the moment and the total payments for the service to the moment are determined, and if the difference between these two values exceeds the minimal payment, then the sum rounded to the multiple of the minimal payment is charged off the user's account.

If the number of charge-offs per week is not set in the accounting period properties, the period is divided into equal parts anyway, but the number of parts is given by the `flow_discounts_per_period` system parameter (by default 64). At the end of each part the event handler runs through all periodic services linked to this period and having flow method of charging. For every such service, the amount to be charged is determined, and if it exceeds the `discount_barrier` parameter, the charge-off is made.

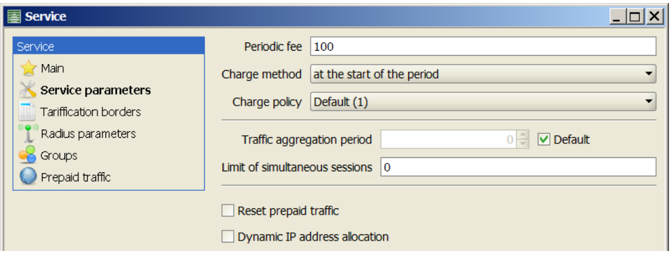
  - **Start date –** is the date when the provision of service begins. Used for purely information purposes.
  - **End date –** is the optional date when the service is shut down and removed altogether, unless it has some service links attached to it.

All these parameters are also relevant for services of other types, which however possess a number of extra parameters specific to each type.

Parameters of the corresponding service link are described in **Periodic service link** on page **110**.

IP traffic service

IP traffic service properties window contains the following pages:



Periodic fee: 100

Charge method: at the start of the period

Charge policy: Default (1)

Traffic aggregation period: 0 ☑ Default

Limit of simultaneous sessions: 0

☐ Reset prepaid traffic

☐ Dynamic IP address allocation

- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
  - **Traffic aggregation interval –** is similar in meaning to the global interface parameter called `traffic_aggregation_interval` (see **Interface parameters** on page **152**), only limited to this service alone.
  - **Limit of simultaneous sessions –** is the limiting number of simultaneous sessions. It is only set when the service is created and it can't be changed later.
  - **Reset prepaid traffic –** is the flag of resetting the prepaid traffic at the end of the accounting period. If set, the unused prepaid traffic is cast to zero, otherwise it is transferred to the next period.
  - **Dynamic IP address allocation –** is used to link this service to another one of hotspot or dialup type (also having this flag set) in order to perform combined tariffication based together on connection time and traffic consumption. One account may have only one IP traffic service with this flag set.
- **Tariffication borders –** contains the tariffication borders, which are the traffic volume values defined for different traffic classes separately and used to set variable traffic prices per megabyte depending on the amount consumed within the accounting period. Each border specification consists of the following values:
  - Traffic class (or a group of classes, see below);
  - Traffic volume;
  - Price.

  There is a default (hidden) border at traffic volume 0 having price 0.
- **RADIUS parameters –** (see **UTM5 RADIUS** on page **159**).
- **Groups –** contains the interface of group tariffication. Group is a union of several traffic classes. The use of groups alters the very logic of tariffication in a following way. Each group is characterized by type, which may be either `max` or `sum`. A group of type `max` is tariffed by the prevailing class, i.e. the price for the whole group is determined via tariffication borders

based on the amount of traffic for the class with maximum traffic. A group of type `sum` is tariffed by the price determined according to the summary amount of traffic for all classes.

Groups must be created ahead of the tariffication borders. Once the custom borders are defined, the group creation interface is disabled.

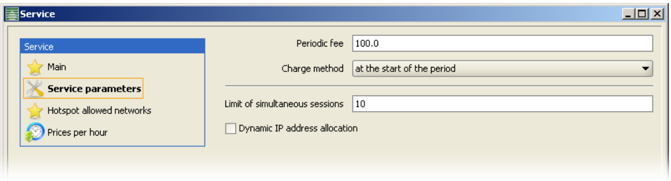
- **Prepaid traffic –** contains the list of prepaid traffic amounts for each of the traffic classes. The prepaid traffic is expended first of all and tariffed by zero price. At the end of the accounting period the unused prepaid traffic is transferred to the following period. The **Accumulate no more than** parameter, if set to a non-zero value, limits the amount of unused traffic that may be transferred so, regardless of its origin. If the parameter is set to 0, accumulation of prepaid traffic is not limited.

Prepaid traffic and group tariffication are mutually exclusive options, i.e. their simultaneous use is impossible.

Parameters of the corresponding service link are described in **IP traffic service link** on page **111**.

Hotspot service

Hotspot service properties window contains the following pages:



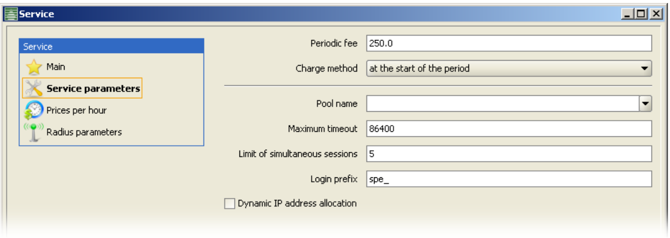
Periodic fee: 100.0
Charge method: at the start of the period
Limit of simultaneous sessions: 10
☐ Dynamic IP address allocation

- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
  - **Limit of simultaneous sessions –** sets the maximum number of concurrent connections that may be established with the same login.
  - **Dynamic IP address allocation –** is used to link this service to another one of IP traffic type (also having this flag set) in order to perform combined tariffication based together on connection time and traffic consumption. One account may have only one hotspot or dialup service with this flag set.
- **Allowed networks –** contains the list of allowed IP addresses from which the user is allowed to authorize on the UTM5 web interface. Authorization requests from other addresses are denied.
- **RADIUS parameters –** (see **UTM5 RADIUS** on page **159**).
- **Price per hour –** contains the list of connection time prices for various time ranges.

Parameters of the corresponding service link are described in **Hotspot service link** on page **114**.

Dialup service

Dialup service properties window contains the following pages:



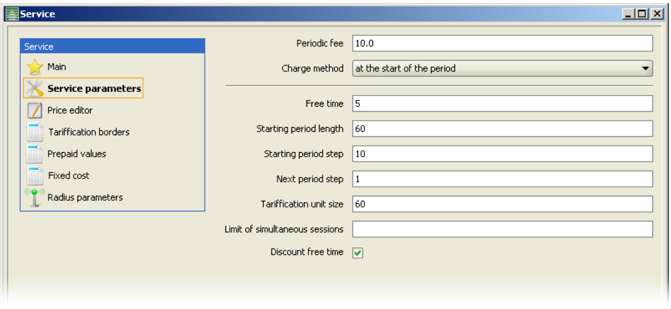
Periodic fee: 250.0
Charge method: at the start of the period
Pool name:
Maximum timeout: 86400
Limit of simultaneous sessions: 5
Login prefix: spe_
☐ Dynamic IP address allocation

- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
  - **Pool name –** is the name of the pool (see **IP Pools** on page **92**) to issue the addresses from. If the pool is registered in UTM5, the first available IP address from it is issued. Otherwise the pool name itself is passed instead.
    This parameter is cached by UTM5 RADIUS.
  - **Maximum timeout –** is the maximum session duration until forced break (in seconds).
  - **Limit of simultaneous sessions –** sets the maximum number of concurrent connections that may be established with the same login.
    This parameter is cached by UTM5 RADIUS.
  - **Login prefix –** is the prefix to be prepended automatically to the user’s login on creation of a service link.
  - **Dynamic IP address allocation –** is used to link this service to another one of IP traffic type (also having this flag set) in order to perform combined tariffication based together on connection time and traffic consumption. One account may have only one hotspot or dialup service with this flag set.
- **Price per hour –** contains the list of connection time prices for various time ranges. Must contain at least one entry. Time is counted with precision to seconds.
- **RADIUS parameters –** (see **UTM5 RADIUS** on page **159**).

Parameters of the corresponding service link are described in **Dialup service link** on page **114**.

Telephony service

Telephony service properties window contains the following pages:



Periodic fee 10.0
Charge method at the start of the period
Free time 5
Starting period length 60
Starting period step 10
Next period step 1
Tariffication unit size 60
Limit of simultaneous sessions
Discount free time ☑

- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
    - **Free time –** is the time threshold (say, 5 sec.) for free calls. Longer calls are charged for based on the full time of the call.
    - **Starting period length –** is the length of initial period having special rounding step.
    - **Starting period step –** is the rounding step for the starting period.
    - **Next period step –** is the rounding step for the rest of the call.
    - **Tariffication unit size –** is the size of time unit to set the price for.
    - **Limit of simultaneous sessions –** sets the maximum number of concurrent connections that may be established with the same login.
    - **Discount free time –** flag, when checked, makes the system consider the prepaid time (see **Prepaid values** on page **68**) in the cumulative summary duration of calls on which the call price per minute may depend.
- **Price editor –** contains the list of call time prices defined separately for various time ranges and for various telephone zones and/or directions.

To create a telephony service, at least one telephone zone or direction must exist in the system.

The prices may be updated all at once according to an arbitrary formula. For example, to mutiply the prices by 1.1, select the rows in question, enter the formula `x*1.1 + 0`, and press **Enter**.

- **Tariffication borders –** contains the tariffication borders, which are the values of total duration of calls per accounting period defined separately for different telephone zones and/or directions and used to set variable call prices depending on the summary duration of calls within the accounting period.

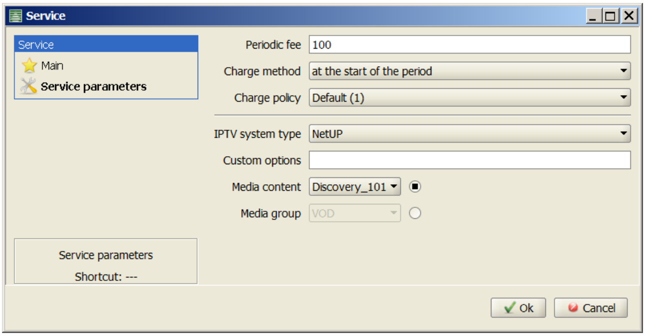
To set the tariffication borders for a particular telephone zone or direction, this zone or direction must be present in the price editor (see above) of this telephony service.

- **Prepaid values –** contains the amounts of prepaid telephone traffic allocated separately for different telephone zones and/or directions.
- **Fixed cost –** contains the fixed cost of a call defined separately for different telephone zones and/or directions. This cost is imposed on every call, regardless its duration, in addition to the variable part defined in **Price editor** and **Tariffication borders**.
- **RADIUS parameters –** (see **UTM5 RADIUS** on page **159**).

Parameters of the corresponding service link are described in **Telephony service link** on page **115**.

IPTV service

IPTV service becomes available after purchasing IPTV integration module license. It’s properties window contains the following pages:



Periodic fee: 100
Charge method: at the start of the period
Charge policy: Default (1)
IPTV system type: NetUP
Custom options:
Media content: Discovery_101 ●
Media group: VOD ○

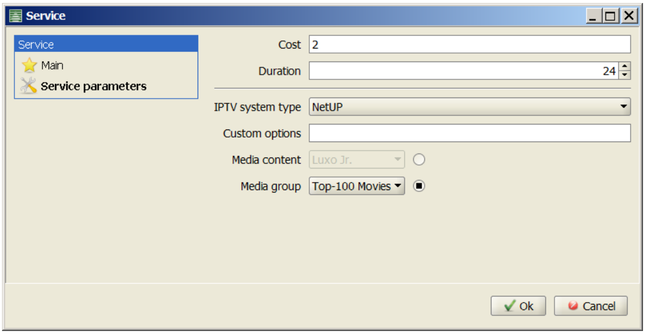
- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
  - **IPTV system type –** is the type of the IPTV system that UTM5 is used with. By default it is set to NetUP and that way the service is configured to work with NetUP IPTV system. That makes available two other parameters - Media content, which allows one to grand access to a certain media content and Media group which allows one to grant access to a group of media contents. For more information on media content and media groups, see NetUP IPTV Administrator’s guide. When this parameter is set to Other, the only available parameter is Custom options.
  - **Custom options –** is custom options that are passed to a corresponding UTM5 RFW event. It is an arbitrary text field that may contain a string of data.

  - **Media content –** is media content that the user will be granted access to when this service will be attached to her account.
  - **Media group –** is media group that the user will be granted access to when this service will be attached to her account.

The corresponding service link parameters may be found in **IPTV service link** on page **116**.

Video on demand service

Video on demand (VoD) service becomes available after purchasing IPTV integration module license. It’s properties window contains the following pages:



Cost 2
Duration 24
IPTV system type NetUP
Custom options
Media content Luxo Jr. ○
Media group Top-100 Movies ●

- **Main –** (standard, **Above**).
- **Service parameters –** is similar to the same page in **Periodic service** (page **62**). In addition, contains the following elements:
  - **IPTV system type –** is the type of the IPTV system that UTM5 is used with. By default it is set to NetUP and that way the service is configured to work with NetUP IPTV system. That makes available two other parameters - Media content, which allows one to grand access to a certain media content and Media group which allows one to grant access to a group of media contents. For more information on media content and media groups, see NetUP IPTV Administrator’s guide. When this parameter is set to Other, the only available parameter is Custom options.
  - **Custom options –** is custom options that are passed to a corresponding UTM5 RFW event. It is an arbitrary text field that may contain a string of data.
  - **Media content –** is media content that the user will be granted access to when this service will be attached to her account.
  - **Media group –** is media group that the user will be granted access to when this service will be attached to her account.

The corresponding service link parameters may be found in **Video on demand service link** on page **116**.

Service templates

The **Service templates** page contain the list of registered service templates (see **Basic system objects: Service templates** on page **33**) with the interface for creating, removing, or editing them.

A service template can be removed only if it is not used at the moment. Otherwise, in the first place it is necessary to remove all services derived from it.

Service templates serve solely as the parent entities for the services included in tariff plans. The properties of a service template are essentially similar to those of a service of the same type (see **Services** on page **61**), except for:

- **Attach by default –** means that the service will be linked to the customer as soon as the tariff link is created
- **Allow multiple linking –** allows one to link the same service to the same customer multiple times

Time ranges

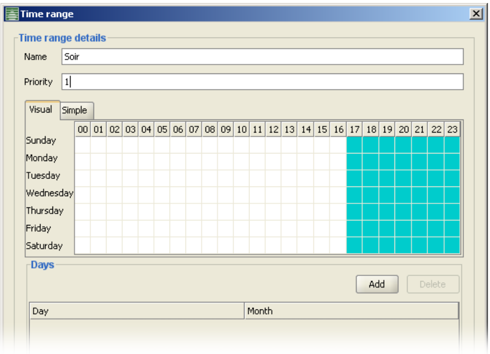
This page contains the list of registered time ranges (see **Basic system objects: Time ranges** on page **37**) with the interface for creating, removing, or editing them.

A time range can be removed only if it is not used at the moment. Otherwise, in the first place it is necessary to remove all entities (services, tariff plans, accounts, etc.) relying on it.

Below is the list of time range parameters and their meanings:

- **ID –** number (assigned automatically).
- **Name –** of the range.
- **Range –** is the set of time intervals that actually constitute the range.
- **Priority –** is the order of precedence of time ranges in case they overlap (see below). The higher the number in the Priority field, the higher priority the time range has.

The Add and Edit buttons open the **Time range** window containing two tabs: **Visual** and **Simple**. On the **Visual** tab a visual editor of time ranges is presented.



Time range

Time range details

Name: Soir

Priority: 1

Visual | Simple

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sunday																		■	■	■	■	■	■	■
Monday																		■	■	■	■	■	■	■
Tuesday																		■	■	■	■	■	■	■
Wednesday																		■	■	■	■	■	■	■
Thursday																		■	■	■	■	■	■	■
Friday																		■	■	■	■	■	■	■
Saturday																		■	■	■	■	■	■	■

Days

Day	Month

Colored cells correspond to the hours included in the range. To include or exclude a single cell, click on it once. With **Shift** key pressed the selection works in block mode, i.e. encloses the whole block once its opposite angles are marked by two consecutive mouse clicks. With **Ctrl** key pressed the selection works in intersection mode, i.e. spans onto cells coinciding by one coordinate with the cell being selected, and by the other coordinate with those selected earlier.

The **Simple** tab presents text interface for setting time ranges. By pressing **Add** and **Remove** enter time ranges for whatever days of week you need.

Visual interface is restricted to hourly precision, i.e. treats each hour as a whole; more accurate timing is possible only within the text interface. Once the fractional hour borders get involved, the visual interface is rendered view-only.

In the **Days** field you may enter individual days to be included in the range. These days are included as a whole, from 12:00 a.m. till 11:59 p.m. If the list of days and the table of hour ranges are used simultaneously, the time range consists of their union.

If the two time ranges overlap, the ambiguous time is appropriated to the one with higher priority. In the case of equal priority the outcome is platform-dependent and generally unreliable. Such collisions are to be avoided.

Reference book

Payment methods

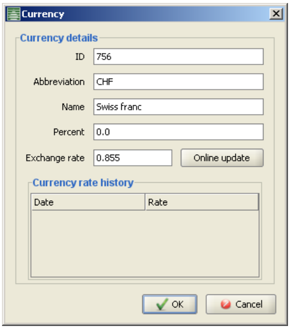
This page contains the list of payment methods (see **Basic system objects: Payments** on page **38**) used to make payments.

The predefined payment methods (those with ID<100) are not editable. User methods are automatically given sequential IDs starting from 100. These methods are functionally equivalent to the predefined **Cash payment** method. User methods are editable, but can not be removed.

Currency

This page contains the list of registered currencies (see **Basic system objects: Currencies** on page **38**) with the interface for creating, removing, or editing them. The Add and Edit buttons open the **Currency details** window.

Currency is characterized by the following parameters:



ID 756
Abbreviation CHF
Name Swiss franc
Percent 0.0
Exchange rate 0.855

Date	Rate

- **ID –** is the digital currency code according to ISO 4217.
- **Abbreviation –** is the three-letter currency code according to ISO 4217.
- **Name –** is the full name of the currency.
- **Percent –** is an arbitrary correction to the exchange rate which is applied upon its online update.
- **Exchange rate –** is the exchange rate against the internal system currency.

  The **Online update** button evokes the online update of the currency exchange rate. This functionality is available only if the `system_currency` (see **Parameters** on page **82**) is set to Russian ruble (ISO code 810), which is the default setting. Otherwise the rate has to be set manually.

This shall not be confused with the standard Refresh *button next to the list of currencies, which refreshes only the list of currencies, but not their rates.*

- **Currency rate history –** contains the history of former exchange rates in the system.

IP zones

List of IP zones is intended for operations with large multi-segment and distributed networks, and contains data on various network segments, including network, mask, and gateway. An IP zone may consist of one or several segments.

IP zones can not be removed.

Buildings

List of connected buildings is intended for convenient operations with networks that span several buildings. The record on house contains its address and the ID of its IP zone.

Buildings can not be removed.

Users may be connected to houses (see **User: Contacts** on page **46**). An example is given in **Linking a user to a house** on page **138**. When a user is connected to a house, the IP addresses for the user are issued from the IP zone associated with the house, see **IP traffic service link** on page **111**.

Banks

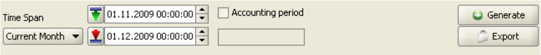
List of banks serves to quickly fill the bank data in the forms. Each record contains the bank ID, BIC, name, address and correspondent account.

Users may be connected to banks (see **User: Additional** on page **46**). An example is given in **Linking a user to a bank** on page **139**.

Reports

UTM5 supports a variety of reports (see the list below). Reports can be generated either for a certain user or for all users at once. Generated reports can be saved as an external file of XML or CSV format.

It is possible to set any time interval for a report to be created. The interval is either entered manually or set to one of the accounting periods.



Time Span: Current Month

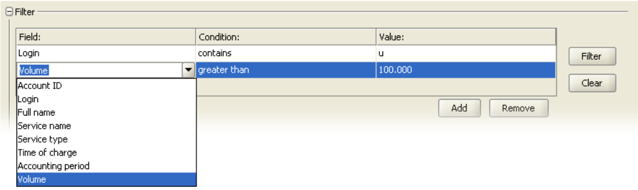
01.11.2009 00:00:00

01.12.2009 00:00:00

☐ Accounting period

When a report for all users is composed, it may be limited to some group of users by selecting the group from the drop-down list. When a report for one user is composed, it may be limited to one of the user's accounts.

The **Filter** roll-up pane may be used for additional filtering of the report data. The exact list of available filtering parameters depends on the report type. An arbitrary set of conditions may be specified and logically combined as either **All conditions** or **Any condition**.



Filter

Field:	Condition:	Value:
Login	contains	u
Volume	greater than	100.000

If the number of entries in the report exceeds 9999, a warning message shows up during its generation. In this case it may be worthwhile to interrupt the operation and generate the report again with narrower sampling, to avoid memory overflow.

Any report contains the summary line that sums up the values across each column.

General report

General report (turnover balance sheet) summarizes all flow of funds on the user's personal accounts during the given time span.

The report includes the following data:

- Personal account ID;
- Initial balance;
- One-time service charges;
- Periodic service charges;
- IP traffic service charges;
- Hotspot service charges;
- Dial-up access service charges;
- Telephony service charges;
- Tax amount;
- Total sum adjusted for taxation;
- Total sum of charges;
- Closing balance.

The general report does not contain the charge-offs caused by the nullification of one’s balance at the end of accounting period (if this option is employed), those caused by the expiration of expiring payments, and the credit payments.

If the selected time span contains no flow of funds, then all columns, including the incoming and closing balance, will read 0.

Traffic report

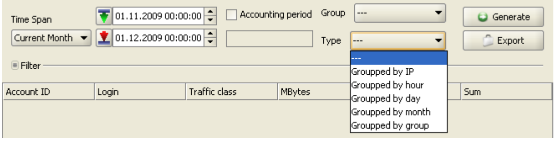
Traffic report summarizes the amount of transferred IP traffic for each personal account and traffic class during the given time span.

Traffic report includes the following data:

- Personal account number;
- Login;
- Number of bytes in a kilobyte;
- Traffic class;
- Amount of traffic transferred (in megabytes);
- Price per unit of transferred traffic (cost of 1 MB);

- Sum charged off the user's personal account.

The data may be grouped by IP addresses, by hours, by days, by months, or by groups of users.



Time Span: Current Month
01.11.2009 00:00:00
01.12.2009 00:00:00
Group: ---
Type: ---

Report on services

This report summarizes information on charge-offs from the user account made during the given time span for the provided services.

The report includes the following data:

- Personal account number;
- Date of charge;
- Accounting period;
- Amount of service provided;
- Service name;
- Service type;
- Comment on the charge.

Besides the total summary line, the lower part of the report contains separate sub-totals on various types of services.

Telephony report

Report on telephony sessions is based on RADIUS server statistics and summarizes data on telephony sessions (calls). The report includes the following fields:

- Session ID;
- Personal account ID;
- Receive date;
- Start date;
- End date;
- Calling station ID;
- Called station ID;
- Access server (NAS) session ID;
- NAS port;
- Login;

- NAS IP address;
- Session status;
- Incoming traffic amount;
- Outgoing traffic amount;
- Telephone zone;
- Telephone direction;
- Duration of the call;
- Rounded duration of the call (calculated based on the rounding step as set in the properties of the telephony service);
- Incoming trunk;
- Outgoing trunk;
- PBX ID;
- Cost per unit time;
- Total cost;
- Disconnect reason.

A call spanning across the border between time periods having different prices per minute is nominally split in two and represented in the report as two calls with the same session ID but with different prices per minute.

A call that has not been tariffed yet is represented in the report as a call with zero price.

Besides the total summary line, the lower part of the telephony report contains sub-totals on individual telephone directions.

Telephony directions report

This report contains data on phone calls aggregated by telephone directions. For each direction it includes:

- Parent telephone zone ID;
- Direction ID;
- Zone name;
- Direction name;
- Total calls duration;
- Total calls cost;
- Number of calls;
- Number of calls with nonzero duration.

Sessions report

Report on modem sessions and VPN sessions is based on RADIUS server statistics and summarizes data on dialup access sessions. The report includes the following fields:

- Session ID;
- Personal account ID;
- Start date;
- End date;
- Calling station ID;
- Called station ID;
- Access server (NAS) session ID;
- IP address issued by NAS;
- NAS port;
- Login;
- NAS IP address;
- Session status;
- Incoming traffic amount (bytes);
- Outgoing traffic amount (bytes);
- Incoming traffic amount (gigabytes);
- Outgoing traffic amount (gigabytes);
- Session duration;
- Termination reason;
- Total cost.

Sessions registered on the RADIUS server (i.e. probably currently active) are highlighted in green.

Report on blockings

This report summarizes information concerning all blockings (see **Accounts** on page **26**) set within the given time span. The following data are available in the report:

- Personal account number;
- Blocking start date;
- Blocking expiry date;
- Blocked item;
- Blocking type;
- Blocking comment.

Report on payments

This report provides information about payments made by a certain user during the given time span. The following data are available in the report:

- Personal account number;
- Actual payment date;
- Date of payment processing by the system;
- Sum of payment in system currency;
- Sum of payment in payment currency;
- Currency of payment;
- Payment method;
- Payer;
- Payment comment.

For the **Credit** payments, the payment status is automatically filled into the **Comment** field:

- **Credit opened till: <date> –** is the status before the expiration date;
- **Credit closed successfully –** is the status after expiration if the account balance is positive;
- **Credit overdue –** is the status after expiration if the account balance is negative.

Besides the total summary line, the lower part of the report contains sub-totals on various methods of payment.

The context menu of the payments report contains the following extra items:

- **Print receipt –** for the given payment;
- **Roll back –** the given payment (not applicable to expiring payments).

Report on expiring payments

This report summarizes the information on expiring payments during the given time span, including the following:

- Account ID;
- Login;
- First payment date;
- Last payment date;
- Payment expiration date;
- Volume;
- Already charged off.

Expired payments are not included in the report.

Other charges

This report contains information about the charge-offs other than service charges, including:

- Expiration of expiring payments;
- Payment rollback;
- Resetting of user's account to zero at the end of period.

In addition, the report contains rebates for the services not used because of blocking, if some of the user's service links have their recalculation options set correspondingly.

Besides the total summary line, the lower part of the report contains sub-totals on various transaction reasons.

Internal transfer

This report provides information about the internal transfers of funds, i.e. the transfers between different accounts of one user, made either via web interface or via `utm5_tray` application.

Report on invoices

This report summarizes the information on invoices issued during the given time span, including the following:

- Internal number;
- External number;
- Account ID;
- Full name;
- Time;
- Paid (invoice status);
- Sum.

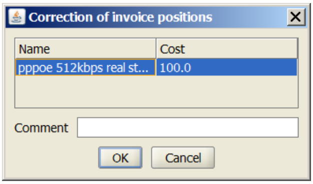
The report for a single user contains an additional button **New invoice** that may be used to generate a new invoice with arbitrary positions. Each position is characterized by:

- Name;
- Quantity;
- Cost per unit;
- Total cost (calculated automatically).

The context menu of the invoices report contains the following extra items:

- **Pay –** opens the payment window (see **Payment page** on page **106**) with payment method set to **Cash** and payment sum set to the sum of the selected invoice;
- **Invoice –** produces the printing version of the selected invoice(s);

- **VAT invoice –** produces the printing version of VAT invoice for the selected item(s);
- **Acceptance report –** produces the printing version of acceptance report for the selected invoice;
- **Send by e-mail –** sends the selected invoice(s) to the e-mail set in the user’s properties.
- **Detail invoice –** generate a detailed invoice with all the services provided. A detailed invoice is mostly used by telephony clients;
- **Correction of invoice positions –** allows one to edit subscriber’s fee positions of the invoice. Any change to invoice positions will affect personal account’s balance in order to preserve the report’s integrity. E.g. when reducing position cost by $5, account’s balance will increase by $5;

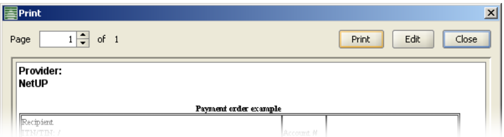


Name	Cost
pppoe 512kbps real st...	100.0

- **Delete invoice –** allows one to delete an invoice.

All changes made via *Correction of invoice positions* appear in the *User change log*.

All invoices, VAT invoices and acceptance reports prepared for printing may be edited (although without saving) by pressing the **Edit** button.



Provider:
NetUP

Payment order example

Recipient
ITN/TIN: /
Account #

User change log

This report lists the changes made to user properties and to some other system objects (services, tariff plans, etc.) during the given time span, including the following information:

- **User ID –** (0 if the change is not related to a user).
- **Login –** (empty if the change is not related to a user).
- **Who –** is the login of the system user responsible for the change.
- **Time –** is the date and time when the change was made.
- **What –** is the short description of the change.
- **Comment**.

Changes of some particular kinds may or may not get included in the report, depending on the way they were made (say, automatically or manually).

Detailed traffic report

This report includes the following detailed information concerning the transferred traffic:

- Date;
- Service link ID;

- Personal account identifier;
- Traffic class;
- IP address and source port;
- IP address and destination port;
- Number of transferred packets;
- Number of transferred bytes;
- TCP flags;
- Protocol;
- TOS.

Detailed statistics over a long time interval typically constitutes huge amounts of data, so the formation of detailed traffic report may take quite a while. In such cases we recommend to use ordinary **Traffic report** instead, or query the database directly using `get_nf_direct`. If creation of a report is taking longer than expected, it may be interrupted (see **Tray icon** on page **116**).

Graphic report

Graphical traffic report provides visual representation of some information depending on the selected **Service type**:

- **IP traffic –** represents the traffic consumption by all users, grouped by traffic classes;
- **Dialup –** represents the number of dialup sessions per hour;
- **Telephony –** represents the number of phone calls per hour.

Custom charges report

This report provides information on the charges performed by third-party systems via integration modules (see **External charges** on page **23**), and includes the following fields:

- Account ID;
- Login;
- Date;
- Mark (a unique ID of the transaction);
- Amount;
- Amount with tax;
- Service;
- Service ID;
- Revoked (a flag that shows whether the charge has been revoked).

DHCP lease

This report provides information on the IP address lease history (IP addresses provided by the UTM5 DHCP module). There are the following columns in the report:

- **ID**
- **IP**
- **MAC**
- **Server id**
- **Client id**
- **Expired** (time and date of lease expiration)
- **Updated** (date and time of lease update)
- **Flags**. Can be the following:
    - **Static –** a static IP address (set manually in IP group settings)
    - **Dynamic –** IP address was assigned automatically
    - **Static, Modified –** a static IP address, the IP group has been modified or deleted since then
    - **Dynamic, Modified –** IP address was assigned automatically, the IP group has been modified or deleted since then

Settings

Parameters

This page contains the system core parameters that are set via Administrator's Interface. The parameters are divided into the following groups:

Tariffication settings

Card user settings

Notification settings

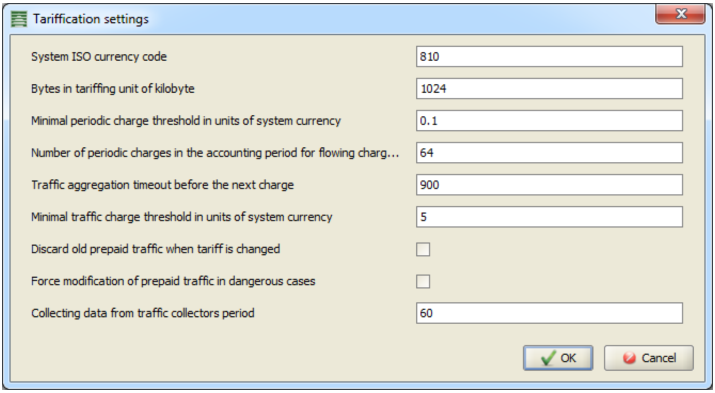
SMTP settings

Invoice document settings

RADIUS protocol settings

Other settings

Double-click a group name or click Edit to open the settings window for the selected settings group.



System ISO currency code	810
Bytes in tariffing unit of kilobyte	1024
Minimal periodic charge threshold in units of system currency	0.1
Number of periodic charges in the accounting period for flowing charg…	64
Traffic aggregation timeout before the next charge	900
Minimal traffic charge threshold in units of system currency	5
Discard old prepaid traffic when tariff is changed	☐
Force modification of prepaid traffic in dangerous cases	☐
Collecting data from traffic collectors period	60

Traffic aggregation timeout before the next charge and Minimal traffic charge threshold in units of system currency determine the charge-off periodicity.

Every charge-off data is added to the charge-off table, which is the most memory-consuming database element. The less the value of the two above-mentioned parameters, the more the table growth rate. In order to reduce the overhead expenses on insert operations with the fastest-growing charge-off table, an archiving function is provided. See ***Archiving of tables*** *on page* ***270****.*

Invoice document settings

Invoice document settings window contains two parameters that determine invoice position aggregation rules. The said are Rules for invoice generating and Rules for prepaid invoice generating.

An aggregation rule consists of comma-separated field names. To have service-related positions included in a single invoice, the field values must be equal for these services.

The following field names are allowed for setting the rule:

tariff.link_id

tariff.id

tariff.name

service.link_id

service.type

service.id

service.name

company.id

company.name

Use colon to separate rules for services that are included in a tariff plan from ones that are not (common services). In this case, if a user have a tariff plan and a common service connected, two invoices will be generated. If a user have only a tariff plan connected, a single invoice will be generated.

The following preset rules could be used as well:

- **default –** predetermined field name set (see description below);
- **single –** all services will be included in a single invoice;
- **separate –** two invoices will be generated: one is for tariff plan services and the other is for common services.

The default rule is equivalent to the following:

```
tariff.link_id,company.id:service.link
```

This means that two invoices will be generated. One will include all positions for the services that are associated with the same tariff plan and the same company. The other will include the positions for each common service.

The single rule is equivalent to empty parameter value field:

```

```

No field names are set, thus all positions will be included in a single invoice.

The separate rule is equivalent to setting colon in the parameter value field:

```
:
```

No field names are set, but the colon determines that all common services will be calculated separately. Hence, if a user have both tariff plan services and common services connected, then two invoices will be generated.

Additional parameters

UTM5 provides the functionality of custom user parameters. Once created, those parameters may be assigned, viewed, and edited via the administrator’s interface along with the rest of user’s properties, as well as used in the document templates. This page contains the interface for creation, modification, and removal of additional users’ parameters.

Parameters’ values for the particular user are set on the user properties page (see **User: Additional** on page **46**). To make a parameter appear there, the **Visible** option has to be checked upon its creation.

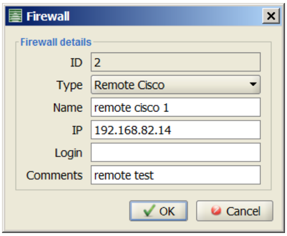
Firewalls

This page contains the list of firewalls registered in the system (see UTM5 RFW on page 165) together with the interface for creating, removing, or editing them. Working firewalls are highlighted in green.

Each firewall has the following parameters:

- **Type –** may be one of the following:
  - **Local –** firewall is used;
  - **Remote Cisco –** firewall controlled via `rsh` is used.

  Firewall type must conform to the `firewall_type` parameter of the config file associated with this firewall.



Firewall details

ID 2

Type Remote Cisco

Name remote cisco 1

IP 192.168.82.14

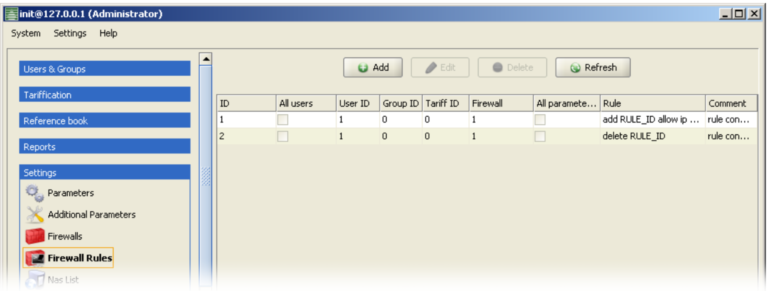
Login

Comments remote test

- **Name –** is the parameter by which the core identifies various rfw. Name must conform to the `rfw_name` parameter of the config file associated with this firewall.
- **IP –** address of the NetFlow supplier stated in the properties of IP traffic service link.
- **Login –** for `rsh` authorization. Valid only for Remote Cisco firewall type; local login is always set to **netup**.
- **Comments –** are optional.

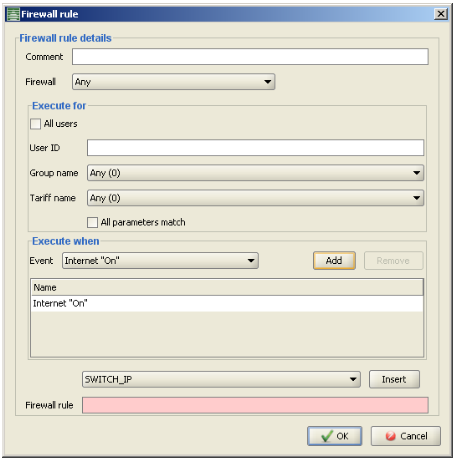
Firewall rules

This page contains the list of firewall rules together with the interface for creating, removing, or editing them.



ID	All users	User ID	Group ID	Tariff ID	Firewall	All paramete…	Rule	Comment
1	☐	1	0	0	1	☐	add RULE_ID allow ip …	rule con…
2	☐	1	0	0	1	☐	delete RULE_ID	rule con…

The Add and Edit buttons open the **Firewall rule** window containing the firewall properties.



Comment:

Firewall: Any

☐ All users

User ID:

Group name: Any (0)

Tariff name: Any (0)

☐ All parameters match

Event: Internet "On"

Internet "On"

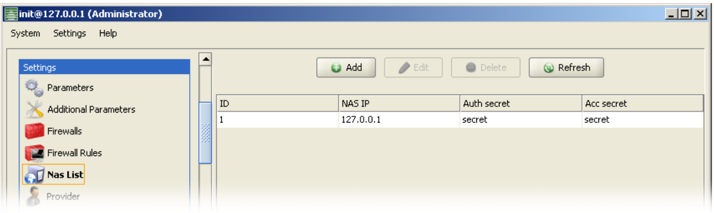
SWITCH_IP

Firewall rule:

The meanings of particular parameters, as well as the general usage of firewall rules, are described in **UTM5 RFW** on page **185**.

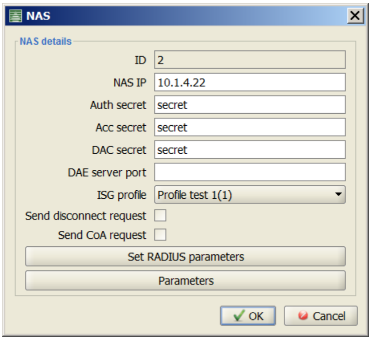
NAS list

The **NAS list** page contains the list of registered network access servers.



ID	NAS IP	Auth secret	Acc secret
1	127.0.0.1	secret	secret

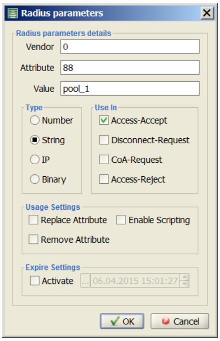
Each NAS is characterized by the following parameters:



ID: 2
NAS IP: 10.1.4.22
Auth secret: secret
Acc secret: secret
DAC secret: secret
DAE server port:
ISG profile: Profile test 1(1)
Send disconnect request ☐
Send CoA request ☐

- **ID –**
- **NAS IP –** (see **IP addresses** on page **40** for the formatting details.)
- **Auth Secret –** is the password used to process authorization requests from NAS and send responses. Must coincide with the common secret word set in the NAS properties for this type of requests.
- **Acc Secret –** is the password used to process accounting requests from NAS and send responses. Must coincide with the common secret word set in the NAS properties for this type of requests.
- **DAC secret –** is the password used in sending extended messages (RFC 5176).
- **DAE server port –** is the port on NAS appointed to receive extended messages.
- **ISG profile –** is an ISG profile that will be used to treat requests coming from this particular NAS (see **ISG profile** on page **89**).
- **Send disconnect request –** flag authorizes the use of Disconnect-request packets on this NAS.
- **Send CoA request –** flag authorizes the use of Change-of-Authorization packets on this NAS.
- **RADIUS parameters –** contains the list of RADIUS attributes linked to the Access-Accept sent to this NAS (for more details see **Additional RADIUS attributes**).

Additional RADIUS attributes



Vendor 0
Attribute 88
Value pool_1

Type: ○ Number ● String ○ IP ○ Binary
Use In: ☑ Access-Accept ☐ Disconnect-Request ☐ CoA-Request ☐ Access-Reject
Usage Settings: ☐ Replace Attribute ☐ Enable Scripting ☐ Remove Attribute
Expire Settings: ☐ Activate 06.04.2015 15:01:27

A list of additional RADIUS attributes for Access-Accept to the given NAS may be set using the **Set RADIUS parameters** button included in NAS properties, service properties, and service link properties.

A RADIUS attribute has the following parameters:

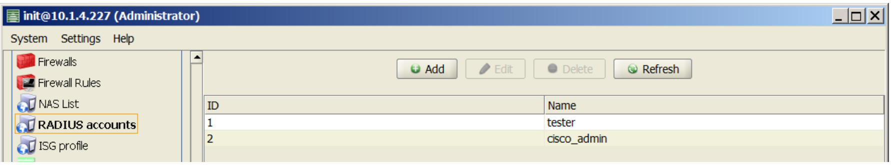
- **Vendor –**
- **Attribute –** name.
- **Value –** of the attribute.
- **Type –** must be either Number, String, IP address, or Binary.
- **Use in –** is the request type to which this attribute should be added.
- **Usage settings –** are the settings that allow one to do the following operations when adding additional attributes to a RADIUS request:
  - **Replace Attribute –** replaces the value of a previously created attribute with the same ID with the value of the current attribute.
  - **Remove Attribute –** removes previously created attribute that has the same ID.
  - **Enable scripting –** allows to use scripting for handling more complex tasks like creating a vendor specific attribute and copying the value of an existing non vendor specific attribute to it.
- **Expire settings –** allows one to set an expire date for this rule. So that these additional attributes will not be added to the RADIUS requests after the expire date.

The additional RADIUS attributes may set some connection parameters (bandwidth limitation, protocol, addresses, etc.) for each service, for a service link, or for a NAS. The access server must be able to support those attributes.

RADIUS attributes are described in RFC 2865 and RFC 2866.

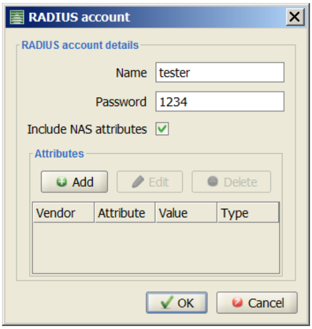
RADIUS accounts

This page contains the list of RADIUS accounts.



ID	Name
1	tester
2	cisco_admin

A RADIUS account allows one to log in on a RADIUS server and receive authorization parameters as a set of attributes that are included in an Access-Accept response type.



Name: tester
Password: 1234
Include NAS attributes ☑

Vendor	Attribute	Value	Type

Press Add to add a new RADIUS account

A RADIUS account has the following set of parameters:

- **ID –** is assigned automatically
- **Name –** is a login name for RADIUS server authorization. It must be unique for the whole UTM5 system
- **Include NAS attributes flag –** makes RADIUS server include the attributes, specified in NAS settings to its response (see **NAS list** on page **87**)

Then comes a table of attributes that need to be included in the RADIUS server response, when one logs in with this account:

- press Add to add an attribute
- press Edit to edit selected attribute
- press Delete to delete selected attribute

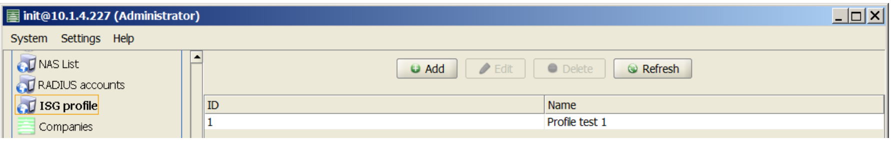
Press Edit on the RADIUS accounts page to edit parameters of the selected account.

Press Delete to delete selected account.

Press Refresh to refresh the list of accounts.

ISG profile

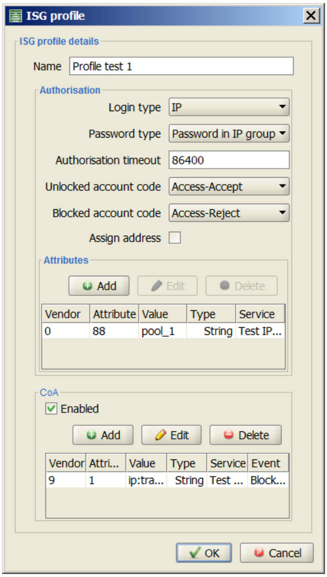
This page contains the list of ISG profiles.



ID	Name
1	Profile test 1

ISG profiles are used for configuring the interaction between RADIUS server and intelligent gateway IPoE (i.e. Cisco ISG). These profiles are used as a parameter for registered NASs.

Press Add to add a new ISG profile. An ISG profile has the following set of parameters.



ISG profile details

Name: Profile test 1

Authorisation

Login type: IP

Password type: Password in IP group

Authorisation timeout: 86400

Unlocked account code: Access-Accept

Blocked account code: Access-Reject

Assign address ☐

Attributes

Vendor	Attribute	Value	Type	Service
0	88	pool_1	String	Test IP...

CoA

☑ Enabled

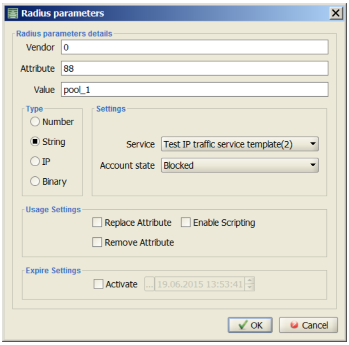
Vendor	Attri...	Value	Type	Service	Event
9	1	ip:tra...	String	Test ...	Block...

- **ID –** is assigned automatically
- **Name –** is an arbitrary information field

Next comes the Authorization parameters group. These parameters determine how to treat authorization requests:

- **Login type –** is the type of data that comes in the User-Name attribute of the authorization request. May be MAC, IP or Login in IP group
- **Password type –** determines if the password should be compared to the password from the IP group properties or with a static password (in that case a *Password* field will appear)
- **Authorization timeout –** is time in seconds to wait for the first packet with an Accounting-Start code. If such a packet doesn’t come before timeout, RADIUS server finishes the session
- **Unlocked account code –** allows one to choose a response code for users whose personal account is not blocked, requesting authorization. The response might be Access-Accept or Access-Reject
- **Blocked account code –** the same as above, but for users whose personal account is blocked
- **Assign address flag –** determines if the Framed-IP-Address attribute should be added to the authorization response. The attribute’s value depends on the login type - MAC or IP. If the authorization type is IP, Framed-IP-address attribute will contain the same IP address that was used for authorization. If the authorization type is MAC, then an IP address determined by the IP group properties will be used (an address from a RADIUS pool, set in the *RADIUS options* tab, or an address from the range, specified in the *Static IP* tab. For more information see **IP groups** on page **111**). If the authorization type is Login in IP group, then IP group settings are used and this flag doesn’t determine whether the Framed-IP-Address attribute will be added or not.

Next comes the Attributes group of parameters, which is a list of additional RADIUS parameters which are added to the authorization response.



Vendor 0
Attribute 88
Value pool_1
○ Number
● String
○ IP
○ Binary
Service Test IP traffic service template(2)
Account state Blocked
☐ Replace Attribute ☐ Enable Scripting
☐ Remove Attribute
☐ Activate 19.06.2015 13:53:41

- press Add to add an attribute

The following parameters are set for a RADIUS attribute:

- **Vendor –** is vendor ID
- **Attribute –** is attribute ID
- **Value –** is the attribute value
- **Type –** is the value type. Can be a number, a string, IP or Binary
- **Settings –** allow one to specify the case when to add this attribute. The attribute may be added depending on the service or on account state (e.g. only when the account is blocked)
- **Usage settings –** allow one to do the following operations when adding an attribute:
  - **Replace attribute –** means that if an attribute with the same ID already exists, its value will be replaced
  - **Remove attribute –** means that if an attribute with the same ID exists, it will be excluded from response
  - **Enable scripting –** allows one to use scripting to solve complicated tasks. E,g, if one needs to create a vendor specific attribute and use a standard attribute value. To do so, fill the Value field with vendor ID and attribute ID, in curly braces, separated by comas (e.g. {9, 44})
- **Expire settings –** allows one to stop adding this attribute after a certain period of time
- press Edit to edit selected attribute
- press Delete to delete selected attribute

Next comes the CoA group of parameters. These parameters determine if CoA requests are applicable and what RADIUS attributes they should contain:

- **Enabled flag –** determines if CoA requests should be used

Settings of an attribute added to a CoA request are similar to settings of an attribute that is added to a response to an authorization request. They allow one to add an attribute depending on the event that caused this CoA request - blocking/unblocking personal account or deleting the service link

- press Edit to edit selected attribute
- press Delete to delete selected attribute

When on *ISG profile* page, select one of the profiles and press Edit to edit it.

Press Delete to delete selected profile.

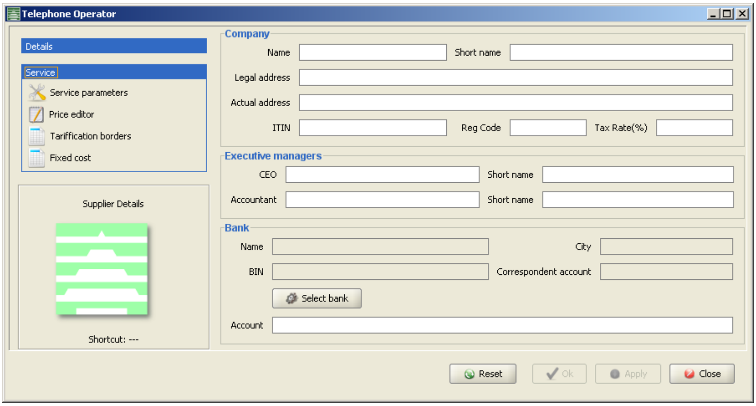
Press Refresh to refresh the list of profiles.

Companies

This page contains legal and financial data on the company's legal entities to use in document templates.

Telephony operators

This page contains the list of telephony operators. These must be stored by UTM5 in order to track mutual settlement charges for the passage of another's telephone traffic. When an operator is created or edited, the following multi-page window shows up:



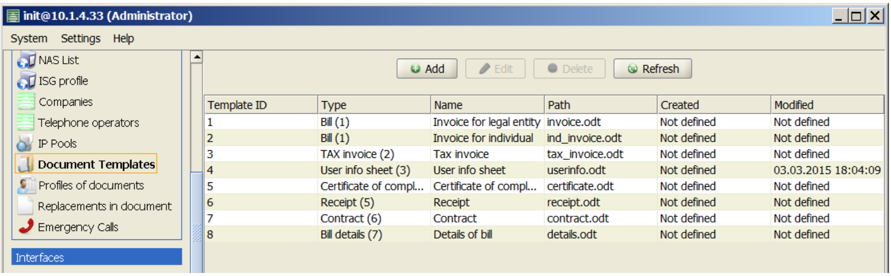
- **Details –** includes the properties of the legal entity identified with this operator, and also the balance of settlement charges.
- **Service –** includes the parameters of the special telephony service intended solely to account for settlement charges with this operator. This service does not show up in the list of services, can not be attached to ordinary users, has no name, no prepaid traffic units, and no linked RADIUS attributes. Tariffication borders for this service are not cumulative, i.e. the service usage amounts on which the price depends refer to one single call. In all other respects it is just like usual telephony service (see **Telephony service** on page **67**).
- **Reports –** (available only when editing an existing operator) includes the reports on payments and charges for this operator.

IP Pools

This page contains the list of pools of IP addresses to be issued to dialup users.

If several IP pools share the same name then the usage thereof is controlled by the parameter named_pool_shuffle (see ***UTM5 RADIUS:*** `named_pool_shuffle` *on page* ***174****). In earlier versions of UTM (prior to 5.2.1-009-update2) the systems's behavior in this case was unpredictable.*

Document templates



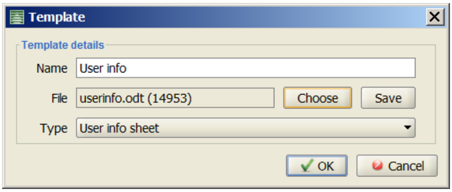
Template ID	Type	Name	Path	Created	Modified
1	Bill (1)	Invoice for legal entity	invoice.odt	Not defined	Not defined
2	Bill (1)	Invoice for individual	ind_invoice.odt	Not defined	Not defined
3	TAX invoice (2)	Tax invoice	tax_invoice.odt	Not defined	Not defined
4	User info sheet (3)	User info sheet	userinfo.odt	Not defined	03.03.2015 18:04:09
5	Certificate of compl...	Certificate of compl...	certificate.odt	Not defined	Not defined
6	Receipt (5)	Receipt	receipt.odt	Not defined	Not defined
7	Contract (6)	Contract	contract.odt	Not defined	Not defined
8	Bill details (7)	Details of bill	details.odt	Not defined	Not defined

This page contains the list of document templates. Basically, a template is an *.odt document that may contain variables (user name, account balance, etc). When a document is being generated from the template, all variables get replaced with their values.

*In case, LibreOffice package is installed on the UTM5 server, documents will be generated in *.pdf format, otherwise they will be generated in *.odt format. Parameters for document generation may be found in the DOCUMENTS section of the UTM5 configuration file (see* ***UTM5 core: Config file*** *on page* ***147****)*

On this page one can do the following:

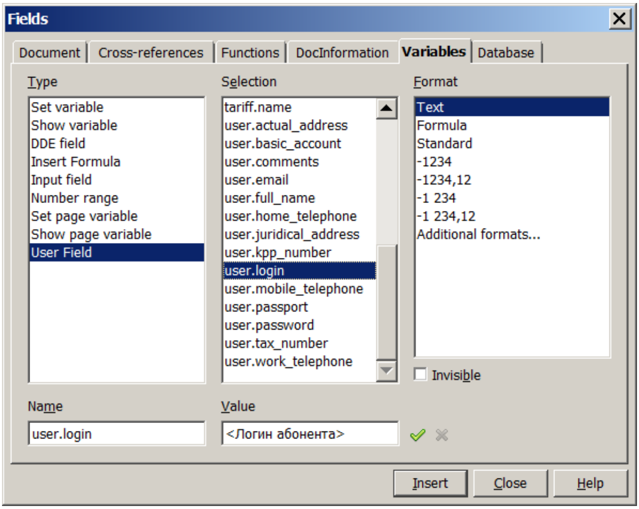
- adding a template
  - press Add to add new document template
  - in the pop-up window press Choose and select the *.odt template file



Name: User info
File: userinfo.odt (14953)
Type: User info sheet

  - then select template type, enter ins name and press Ok
- editing a template:
  - select a template and press Edit
  - press Save in the pop-up window to save the current template to an *.odt file for editing
  - open saved *.odt file in LibreOffice and edit it

- go to *Insert* > *Fields* > *Other* and go to the *Variables* tab

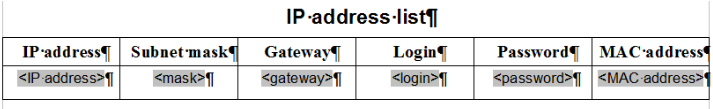


Type: User Field
Selection: user.login
Format: Text
Name: user.login
Value: <Логин абонента>

- choose field type *User Field* and choose a variable from the list or enter it’s name and press **Insert**

Make sure that the variable that you want to insert is available for the type of template you are editing (see

- if a variable returns a list of values, it should be placed in a table row like this:



IP·address·list¶

IP·address¶	Subnet·mask¶	Gateway¶	Login¶	Password¶	MAC·address¶
<IP·address>¶	<mask>¶	<gateway>¶	<login>¶	<password>¶	<MAC·address>¶

The table will be automatically extended with the number of rows to hold all the values of the variable.

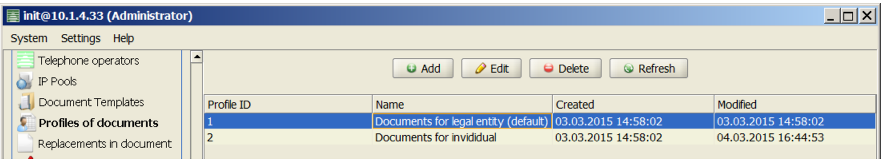
If all the variables in a table row turn out to be empty, the row is deleted

- press Delete to delete selected variable
- press Refresh to refresh the list of templates

The set of available variables for a template depends on its type. Full list of variables is available in **Appendix: Variables** on page **293**.

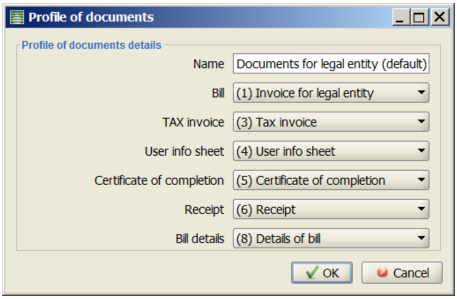
Document profiles

This page contains a list of document profiles.



Profile ID	Name	Created	Modified
1	Documents for legal entity (default)	03.03.2015 14:58:02	03.03.2015 14:58:02
2	Documents for invididual	03.03.2015 14:58:02	04.03.2015 16:44:53

A profile of documents is a set of templates with one template for each document type:



Name: Documents for legal entity (default)
Bill: (1) Invoice for legal entity
TAX invoice: (3) Tax invoice
User info sheet: (4) User info sheet
Certificate of completion: (5) Certificate of completion
Receipt: (6) Receipt
Bill details: (8) Details of bill

These profiles are used for generating documents like contract or an invoice. Each user is assigned one of the profiles.

press Add to add a new profile of documents

press Edit to edit selected profile

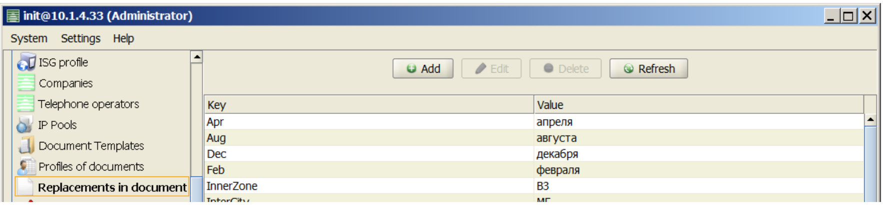
press Delete to delete selected profile

press Refresh to refresh the list of profiles

The first profile (default) can not be deleted. It is set as a default profile for all users. One can assign another profile to a user in User’s properties (User > Other)

Replacements in documents

This page contains the list of replacements that can be used in document templates (**Above**).



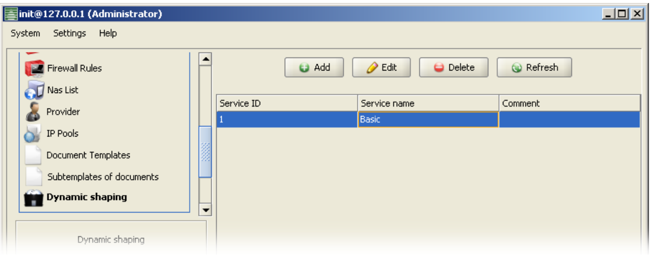
Key	Value
Apr	апреля
Aug	августа
Dec	декабря
Feb	февраля
InnerZone	ВЗ

- press Add to add a new replacement

- select one of existing replacements and press Edit to replace its value
- press Delete to delete a replacement
- press Refresh to refresh the list of replacements

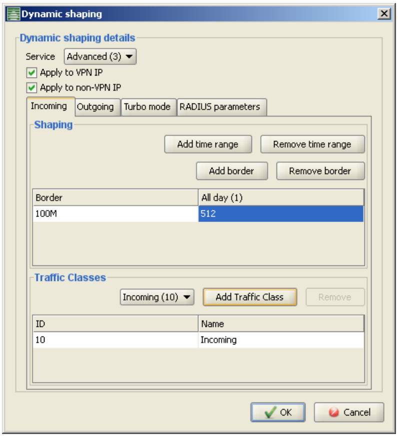
Dynamic shaping

This page displays the dynamic shaping parameters for those services which are enabled with it, together with the interface for enabling or disabling shaping and setting its parameters. Shaping is applicable solely to the IP traffic services.



Service ID	Service name	Comment
1	Basic	

The Delete button disables shaping for the selected service. The Add and Edit buttons open the **Dynamic shaping** window containing the following interface elements:



Service: Advanced (3)
☑ Apply to VPN IP
☑ Apply to non-VPN IP

Border	All day (1)
100M	512

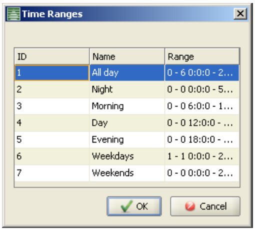
Incoming (10)

ID	Name
10	Incoming

- **Service –** is the drop-down list for selecting the service to apply shaping to. Disabled during editing (may be set only once).
- **Apply to VPN IP, Apply to non-VPN IP –** are the flags controlling the application of shaping to different types of IP addresses. Disabled during editing (may be set only once).
- On the tabs **Incoming** and **Outgoing**:

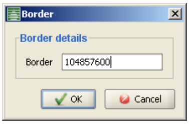
    - Using the buttons **Add time range** and **Remove time range** select the time range to apply shaping.

Usage of the overlapping time ranges is inadmissible here, as well as in other circumstances.



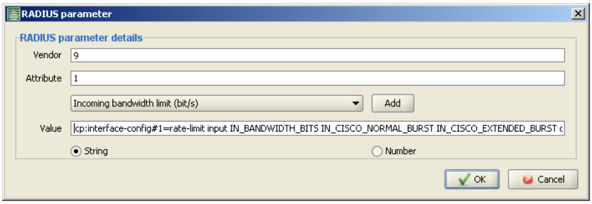
ID	Name	Range
1	All day	0 - 6 0:0:0 - 2...
2	Night	0 - 0 0:0:0 - 5...
3	Morning	0 - 0 6:0:0 - 1...
4	Day	0 - 0 12:0:0 - ...
5	Evening	0 - 0 18:0:0 - ...
6	Weekdays	1 - 1 0:0:0 - 2...
7	Weekends	0 - 0 0:0:0 - 2...

    - Using the buttons **Add border** and **Remove border** set the border values of traffic amount (in bytes) for shaping. The units of K (kilobytes), M (megabytes), or G (gigabytes) may be used; if no units are given, the number is interpreted as a value in bytes. It may be worthwhile to set the lower border to zero, so as to establish bandwidth limitations for any amount of traffic starting from zero.



Border: 104857600

    - For each border and for each time range set the bandwidth limitation. It will be valid during the given time range if the amount of traffic is above the given border (and still below the next one, if any).
    - In the **Traffic classes** group enter the desired traffic classes to apply shaping to, using the drop-down list and the buttons **Add traffic class** and **Remove**.
- On the **RADIUS parameters** tab enter the parameters to be sent to the RADIUS server, including **Vendor**, **Attribute**, and the command itself in the **Value** field. The command may include variables selected from the drop-down list. See **UTM5 Dynashape: RADIUS parameters** on page **200** for the list of variables. On execution the variables are substituted with their values.



Vendor: 9

Attribute: 1

Incoming bandwidth limit (bit/s)

Value: cp:interface-config#1=rate-limit input IN_BANDWIDTH_BITS IN_CISCO_NORMAL_BURST IN_CISCO_EXTENDED_BURST

● String ○ Number

- **Turbo mode** tab is optionally used to set up the turbo mode, which is a brief temporary enhancement of Internet access bandwidth. Its parameters include: bandwidth limitation for incoming and outgoing channels separately (or no limitation whatsoever, if such an option is selected), duration, and the name of a one-time service used as a payment for engaging turbo mode. Customers may switch turbo mode on via web interface at their own discretion.

Emergency calls

This page contains the list of telephone zones and/or directions which are available for a call even when the user's account is blocked.

Archive DB

This page contains an interface for automatic DB archiving

Interfaces

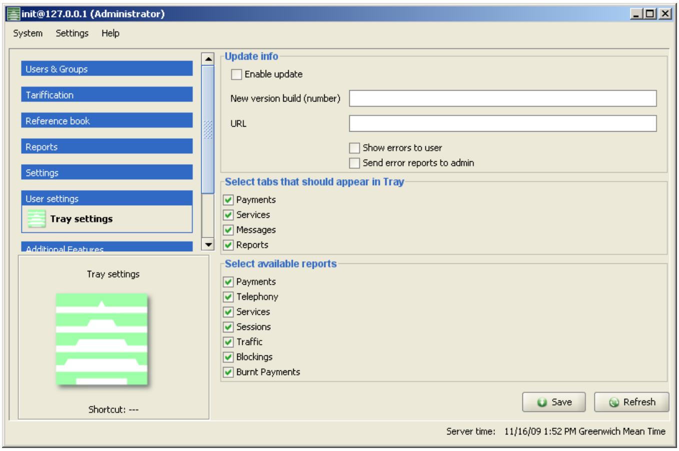
This section covers the settings affecting user’s and cashier’s interfaces. **Tray settings** on page **98** contains settings related to `utm5_tray` application (see **UTM5 tray utility** on page **233**). Parallel functionality of the web interface may be set up by editing the XML templates in the root directory of the interface.

The pages **Tariff switch**, **Voluntary suspension**, **Promised payments**, and **Internal transfer** on page **102** describe each a particular facility defined by a set of parameters. An arbitrary amount of separate sets may be created. Each set has its own priority and refers to certain group of users. The settings of the given facility for the given user are defined by the set that refers to the corresponding group. If there are several sets defined for this group (or the user belongs to multiple groups), only one set is applied, namely the one with the highest priority. When this set is disabled, so is this facility for this user, despite possible presence of other sets, if even for the same group.

Priority of sets may be edited by the buttons **Up** and **Down**. There is one special set associated with the group **All** which is present by default, can not be removed, and has fixed priority of 0 (i. e. below others).

Tray settings

This page controls the settings related to the client tray application (see **UTM5 tray utility** on page **233**).



Update info

☐ Enable update

New version build (number)

URL

☐ Show errors to user

☐ Send error reports to admin

Select tabs that should appear in Tray

☑ Payments

☑ Services

☑ Messages

☑ Reports

Select available reports

☑ Payments

☑ Telephony

☑ Services

☑ Sessions

☑ Traffic

☑ Blockings

☑ Burnt Payments

Shortcut: ---

Server time: 11/16/09 1:52 PM Greenwich Mean Time

The **Update info** group of parameters controls the automatic update settings:

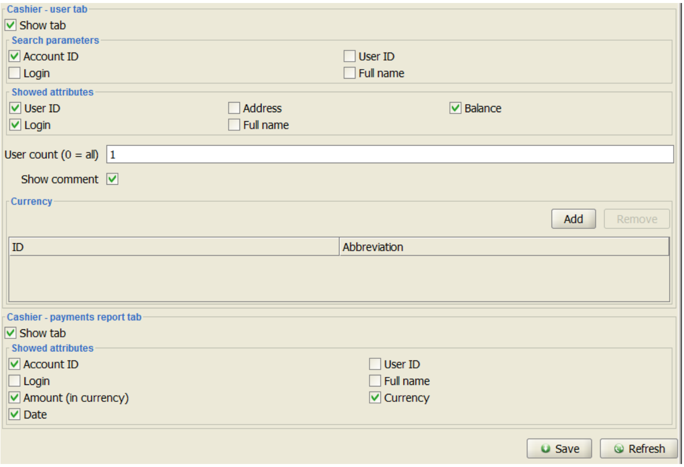
- **Enable update –** switches on automatic update.
- **New version build (number) –** is the build number of new version of the user application. On startup it is compared to the current version number, and once the current version is found to be outdated, it is updated.
- **URL –** is the network address to download the new version from.

The **Select tabs...** group of parameters controls users’ access to particular interface pages (see **UTM5 tray utility: Interface pages** on page **234**).

The **Select available reports** group of parameters controls users’ access to particular kinds of reports on the **Reports** page (see **UTM5 tray utility: Reports** on page **235**).

Cashier interface

This page controls the settings of UTM5 cashier interface (see **cashier module** on page **227**).



Cashier - user tab

☑ Show tab

Search parameters

☑ Account ID ☐ User ID

☐ Login ☐ Full name

Showed attributes

☑ User ID ☐ Address ☑ Balance

☑ Login ☐ Full name

User count (0 = all) 1

Show comment ☑

Currency

ID	Abbreviation

Cashier - payments report tab

☑ Show tab

Showed attributes

☑ Account ID ☐ User ID

☐ Login ☐ Full name

☑ Amount (in currency) ☑ Currency

☑ Date

Settings related to the payment page include:

- **Show tab –** checks whether or not include this page in the cashier’s interface.
- **Search by –** lists the parameters to allow searching by.
- **Attributes to show –** lists the parameters to show in the search results.
- **Users count –** is the number of users to show in the search results.
- **Show comment –** shows the comment field in the cashier’s interface.
- **Currency –** is the list of available currencies.

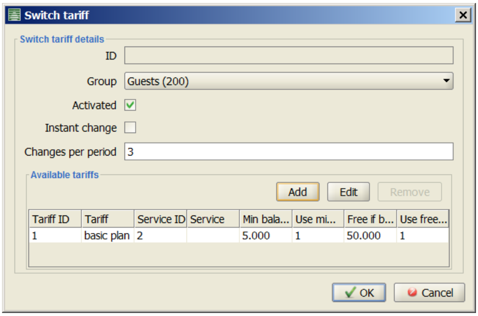
Settings related to the report page include:

- **Show tab –** checks whether or not include this page in the cashier’s interface.
- **Attributes to show –** lists the payment parameters to show in the report.

Tariff switch

This page contains the list of sets of parameters for switching tariff plans. The applicability of sets to a particular user is determined based on group affiliation and priorities (see above).

The Add and Edit buttons open the properties window containing the following elements:



ID:

Group: Guests (200)

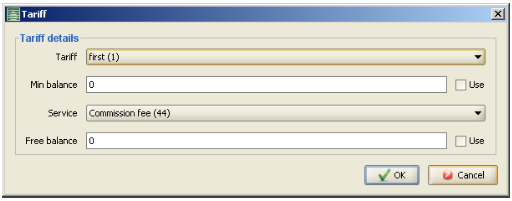
Activated: ☑

Instant change: ☐

Changes per period: 3

Tariff ID	Tariff	Service ID	Service	Min bala...	Use mi...	Free if b...	Use free...
1	basic plan	2		5.000	1	50.000	1

- **ID –** of the set.
- **Group –** of users to which this set is applicable.
- **Activated –** is the flag that enables the set.
- **Instant change –** is the flag that allows the plan to be switched instantly. Otherwise the plan will be switched at the end of the accounting period.
- **Changes per period –** is the maximum allowed number of plan switches during one accounting period.
- **Available tariffs –** is the list of tariff plans eligible for the switch. The list is controlled by the buttons **Add**, **Edit**, and **Remove**.



Tariff: first (1)

Min balance: 0 ☐ Use

Service: Commission fee (44)

Free balance: 0 ☐ Use

Each item in the list has the following properties:

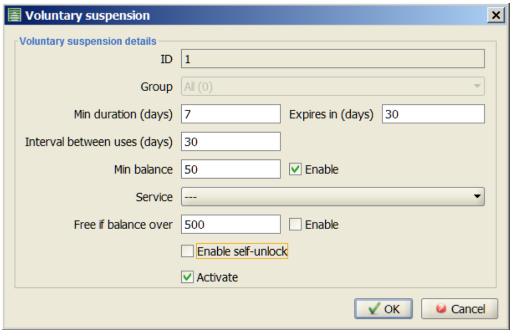
- **Tariff –** is the tariff plan name;
- **Min balance –** is the minimum value of the user's balance required to switch plans;
- **Service –** is a one-time service used to collect fee for the plan switch;
- **Free if balance over –** is the minimum balance value required to switch plans for free.

Voluntary suspension

This page contains the list of sets of parameters for voluntary suspension. The applicability of sets to a particular user is determined based on group affiliation and priorities (see above).

After suspension is activated, recurring fee is recalculated according to the periodic service link parameters (see **Periodic service link** *on page* **110***)*

The Add and Edit buttons open the properties window containing the following elements:



ID: 1
Group: All (0)
Min duration (days): 7
Expires in (days): 30
Interval between uses (days): 30
Min balance: 50 ☑ Enable
Service: ---
Free if balance over: 500 ☐ Enable
☐ Enable self-unlock
☑ Activate

- **ID –** of the set.
- **Group –** of users to which this set is applicable.
- **Min duration –** of the suspension.
- **Max duration –** of the suspension.
- **Interval between uses –** of this facility.
- **Min balance –** is the minimum value of the user's balance required to use suspension.
- **Service –** is a one-time service used to collect fee for the suspension.
- **Free if balance over –** is the minimum balance value required to suspend for free.
- **Enable self-unlock –** is a flag enabling the user to lift the suspension prematurely.
- **Activated –** is the flag that enables the set.

After the voluntary blocking is lifted, the Internet status for the account remains **Off** till the next payment, or till the end of the month, or until turned on by the user, whichever happens sooner. The users may turn it on using the `utm5_tray` application (see **UTM5 tray utility: Main** on page **234**) or via the web interface (see **Web interface: Accounts** on page **241**).

Promised payments

This page contains the list of sets of parameters for making promised payments. The applicability of sets to a particular user is determined based on group affiliation and priorities (see above).

The Add and Edit buttons open the properties window containing the following elements:

- **ID –** of the set.
- **Group –** of users to which this set is applicable.
- **Max payment –** that can be made.

- **Expires in –** (term of expiration in days).
- **Interval between uses –** of this facility.
- **Min balance –** is the minimum value of the user's balance required to make promised payments.
- **Service –** is a one-time service used to collect fee for the promised payments.
- **Free if balance over –** is the minimum balance value required to make promised payments for free.
- **Activated –** is the flag that enables the set.

Internal transfer

This page contains the list of sets of parameters for making internal transfers. The applicability of sets to a particular user is determined based on group affiliation and priorities (see above).

The Add and Edit buttons open the properties window containing the following elements:

- **ID –** of the set.
- **Group –** of users to which this set is applicable.
- **Activated –** is the flag that enables the set.

Additional Features

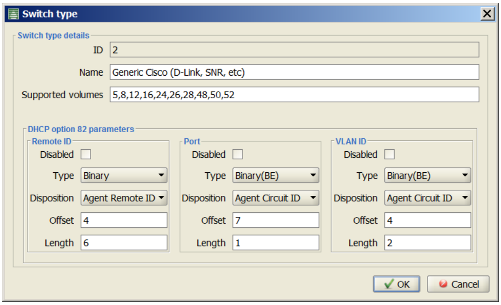
- **Statistics –** summarizes info on uptime and the number of NetFlow records in the system.
- **LibURFA Plugins –** displays the list of plugins and their version numbers.
- **Symbols –** displays the list of URFA functions.
- **Connections –** displays the list of open connections.
- **Hotspot sessions –** displays the list of Hotspot sessions opened via the web interface (but not the RADIUS hotspot sessions).
- **RADIUS attributes –** displays the list of custom RADIUS attributes. These may be attached to particular users, network access servers, services of IP traffic, dialup, and telephony, and also to corresponding service links.
- **RADIUS sessions –** displays the list of active sessions on RADIUS server.
- **Change password –** contains the interface for changing the administrator password. The **Change** button turns active only if **New password** and **Confirm new password** coincide.

Inventory

This group contains pages that allow one to tweak the entities, used by UTM5 DHCP module (see **UTM5 DHCP** on page **247**)

Switch types

This page contains the list of available switch types.



Switch type

Switch type details

ID: 2

Name: Generic Cisco (D-Link, SNR, etc)

Supported volumes: 5,8,12,16,24,26,28,48,50,52

DHCP option 82 parameters

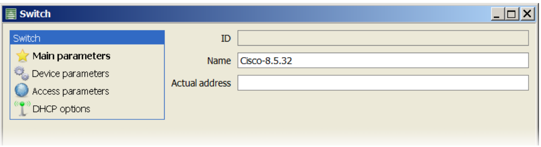
	Remote ID	Port	VLAN ID
Disabled	☐	☐	☐
Type	Binary	Binary(BE)	Binary(BE)
Disposition	Agent Remote ID	Agent Circuit ID	Agent Circuit ID
Offset	4	7	4
Length	6	1	2

Press Add to add a new switch type. It has the following parameters:

- **Name –** is the name for this switch type
- **Supported volumes –** is the number of available ports for this switch type (may be several values, comma separated, e.g. for different switch models)
- **DHCP option 82 parameters –** is a description of the option 82 parameters used to attach an IP address to a switch, acting as a DHCP Relay or to a switch port which the DHCP request came from. DHCP option 82 carries a switch ID (Remote ID), port number and VLAN ID. Each parameter has:
  - **Type –** is a parameter type (string or binary)
  - **Disposition –** is a suboption of the option 82 to which the offset is applied. Suboption code is considered to be the start of the suboption and the start of the offset. If the **Option 82** value is set, the offset starts at the beginning of the whole option 82.
  - **Offset –** is an offset in bytes. It shows the beginning of this parameter in respect to the beginning of the option or one of its parts
  - **Length –** is the length of the parameter in bytes

Switches

This page contains the list of switches available in the system. Every switch description contains its ID, name, comment and other parameters that allow one to identify this particular switch.



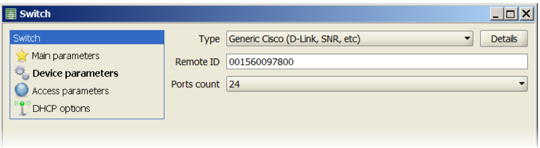
ID:

Name: Cisco-8.5.32

Actual address:

Press Add to add a new switch. When adding a new switch, one has to enter the following parameters:

- **Main parameters**:
  - **ID –** is an internal switch ID (is assigned automatically)
  - **Name –** is the switch name in the database (name uniqueness is not checked, but is recommended)
  - **Actual address –** is a field that contains information about the actual address of this particular switch

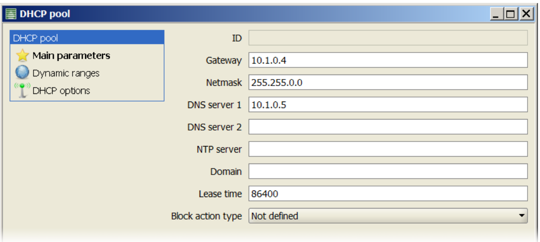


Type: Generic Cisco (D-Link, SNR, etc)
Remote ID: 001560097800
Ports count: 24

- **Device parameters**:
    - **Type –** is a list that allows to choose one of the existing switch types. Press Details to see the parameters of this switch type and to make sure that the switch fits those parameters.
    - **Remote ID –** is a DHCP option 82 parameter Remote ID. It is used by the switch to form DHCP requests. The parameter type is set in the corresponding switch type properties.
    - **Ports count –** is the number of ports in the switch. Choose the appropriate number. This list is created in the switch type properties on the **Switch type** page.
- **Access parameters –** are the switch access parameters. These parameters might be used in firewall rules (RFW module)
- **DHCP options –** are auxiliary DHCP options. If those options are set, they will be included in the DHCP response, if they were present in a corresponding DHCP request

DHCP pools

This page contains the list of the DHCP pools. The connection between a DHCP pool and a DHCP client or a DHCP Relay is set in the service link parameters.



ID:
Gateway: 10.1.0.4
Netmask: 255.255.0.0
DNS server 1: 10.1.0.5
DNS server 2:
NTP server:
Domain:
Lease time: 86400
Block action type: Not defined

Press Add to add a new pool of IP addresses. When adding a new pool, one should set the following parameters:

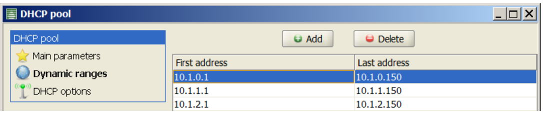
- **Main parameters –** are the basic settings and the block action type. The basic settings include:
    - ID (is assigned automatically)
    - Gateway
    - Netmask
    - DNS server 1
    - DNS server 2
    - NTP server
    - Domain
    - Lease time

The block action type defines the DHCP server behavior in case a DHCP request comes from a blocked user (a user who’s personal account is blocked). This parameter might have the following values:

- **Not set –** which means that the DHCP server will lease an IP address for both blocked and not blocked users

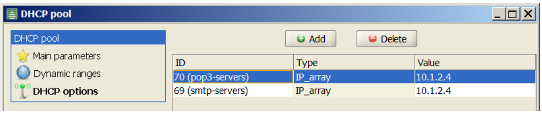
  - **Use blocked –** is an option that allows one to assign IP addresses from a certain DHCP pool to blocked users. In order to use this option, one has to select a pool that will be used for treating requests from blocked users. This option is only available when more than one DHCP pool is registered in the system
  - **Ignore request –** means that DHCP requests coming from blocked users will be ignored by the DHCP server

- **Dynamic ranges –** is the range of addresses for the current DHCP pool.



First address	Last address
10.1.0.1	10.1.0.150
10.1.1.1	10.1.1.150
10.1.2.1	10.1.2.150

- **DHCP options –** is a set of auxiliary DHCP options. If those options are set, they will be included in the DHCP response, if they were present in a corresponding DHCP request.

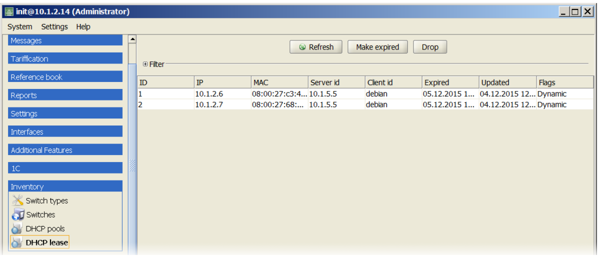


ID	Type	Value
70 (pop3-servers)	IP_array	10.1.2.4
69 (smtp-servers)	IP_array	10.1.2.4

DHCP lease

This page contains the list of active and expired DHCP leases.

The table has the following columns:



ID	IP	MAC	Server id	Client id	Expired	Updated	Flags
1	10.1.2.6	08:00:27:c3:4...	10.1.5.5	debian	05.12.2015 1...	04.12.2015 12...	Dynamic
2	10.1.2.7	08:00:27:68:...	10.1.5.5	debian	05.12.2015 1...	04.12.2015 12...	Dynamic

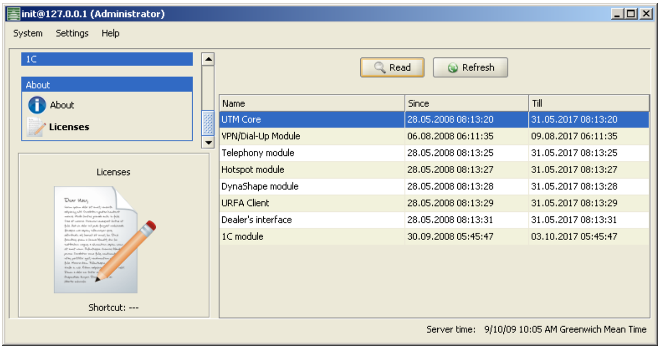
- **ID –** is an automatically assigned record number
- **IP –** is the leased IP address
- **MAC –** is the client’s MAC address
- **Server ID –** is the server’s IP address
- **Client ID –** is a HostName attribute of the DHCP option 12 from the client’s DHCP request
- **Expires –** is the date of the IP address lease expiration
- **Updated –** is the start date of the IP address lease

About

This group contains two pages:

- **About –** contains the program version number and contact info of the license holder.

- **Licenses –** displays list of separate modules’ licenses and their terms of validity. Pressing **Read** reveals specific inner parameters and limitations of a particular module.



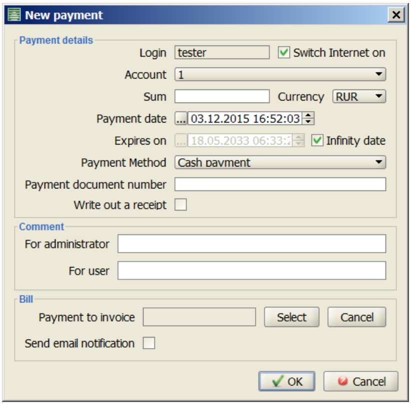
Name	Since	Till
UTM Core	28.05.2008 08:13:20	31.05.2017 08:13:20
VPN/Dial-Up Module	06.08.2008 06:11:35	09.08.2017 06:11:35
Telephony module	28.05.2008 08:13:25	31.05.2017 08:13:25
Hotspot module	28.05.2008 08:13:27	31.05.2017 08:13:27
DynaShape module	28.05.2008 08:13:28	31.05.2017 08:13:28
URFA Client	28.05.2008 08:13:29	31.05.2017 08:13:29
Dealer's interface	28.05.2008 08:13:31	31.05.2017 08:13:31
1C module	30.09.2008 05:45:47	03.10.2017 05:45:47

Server time: 9/10/09 10:05 AM Greenwich Mean Time

Stray pages

Payment page

The payment page opens in a separate window and contains the following interface elements:



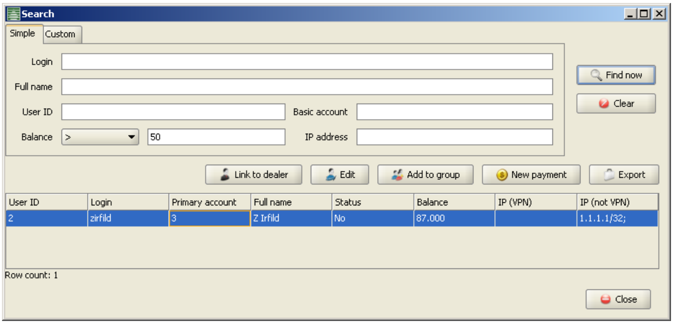
Login: tester
☑ Switch Internet on
Account: 1
Sum:
Currency: RUR
Payment date: 03.12.2015 16:52:03
Expires on: 18.05.2033 06:33:2
☑ Infinity date
Payment Method: Cash payment
Payment document number:
Write out a receipt ☐
For administrator:
For user:
Payment to invoice:
Send email notification ☐

- **Login –** is the user login (view-only).
- **Switch internet on –** enables switching the Internet status for the given account to **On** after making the payment, should the resulting account balance be positive. The default setting for this option (`on` / `off`) is defined by one of the interface parameters, see **Settings: TurnInternetOn** on page **42**.
- **Account –** is a drop-down list for selecting an account, if the user has more than one of them.
- **Sum –** of the payment.
- **Currency –** of the payment.
- **Payment date –** is the date when the payment is made.
- **Expires on –** is the date of expiration of the payment (optional parameter).
- **Payment method –** is a drop-down list for selecting one of the registered payment methods. Once the payment method is set to **Loan**, the **Expires on** parameter becomes mandatory.
- **Payment document number –** is the number of external document (if any) being the reason for the payment.
- **Write out a receipt –** enables the generation of a receipt for printing on pressing ✓ Ok.
- **Comment (For administrator, For user) –** are arbitrary comments.

- **Payment to invoice –** is the number of internal invoice (if any) being the reason for the payment. If the payment is being made according to some internal invoice, the **Sum** and **Currency** fields get filled with the values from the said invoice and turn inactive.
- **Send email notification –** enables sending an e-mail notification to the user.

Search page

The search page opens in a separate window and contains the following interface elements:



Login
Full name
User ID
Basic account
Balance: >   50
IP address

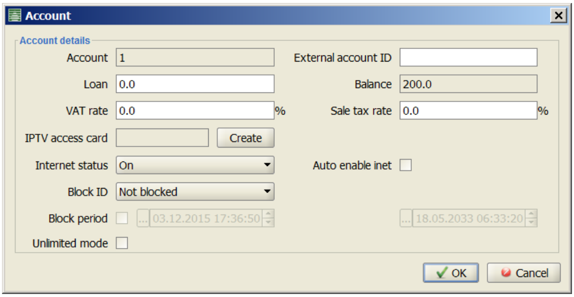
User ID	Login	Primary account	Full name	Status	Balance	IP (VPN)	IP (not VPN)
2	zirfild	3	Z Irfild	No	87.000		1.1.1.1/32;

Row count: 1

- On the **Simple** tab one may perform a text search by login or by full name, as well as search by user ID, primary account ID, IP address, or account balance.
- On the **Custom** tab one may perform a search by combination of an arbitrary number of conditions including any of the user’s properties.
- The **Link to dealer** button opens the interface for linking the user to a dealer, like the similar button on the user properties page (see **User: Main** on page **45**).
- The **Edit** button opens the user properties window, like the similar button on the user list page (see **Users** on page **44**).
- The **Add to group** button opens the interface for adding selected users to a group, like the similar button on the user properties page (see **User: Groups** on page **46**).
- The **New payment** button opens the payment window (see **Payment page** on page **106**).

User account

User account is an object containing the financial information. User accounts are created, modified or deleted via the user details window (see **Tariffication: Accounts** on page **46**). A user may have multiple accounts.



Account: 1
External account ID:
Loan: 0.0
Balance: 200.0
VAT rate: 0.0 %
Sale tax rate: 0.0 %
IPTV access card:
Internet status: On
Auto enable inet: ☐
Block ID: Not blocked
Block period: ☐ 03.12.2015 17:36:50   18.05.2033 06:33:20
Unlimited mode: ☐

User account has the following parameters:

- **Account –** ID (is assigned automatically).
- **External account ID –** (optional) for integration with some external system.
- **Loan –** of the account (may be changed either manually in this window, or by making payments with special method called **Credit**).
- **Balance –** of the account (view-only).
- **VAT rate –** to be applied before invoicing.
- **Sale tax rate –** to be applied before invoicing.
- **IPTV access card –** is the user account IPTV access card number. This field might be empty if the card hasn't been generated yet. In that case press Create to generate an IPTV access card for this user account.
- **Internet status –** of the account (on / off). The Auto enable inet flag allows one to set up UTM5 to automatically enable Internet for this personal account as soon as the customer replenishes her account and the account unblocks.

The flags Don't charge recurrent fee and Decrease prepaid traffic which were among the account properties in UTM versions up to 5.2.1-008 have been since transferred to charge policy properties, see ***Charge policy*** *on page* ***35****.*

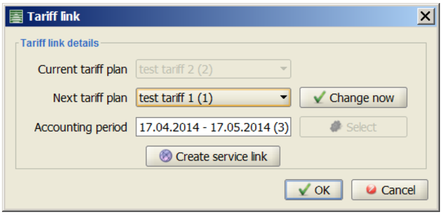
- **Block type –** shows whether the account is blocked, which can be done arbitrarily by the administrator or automatically by the system. Note that after lifting a manual blocking you have to turn Internet on for that account (this is done in the same window, see **Internet status** above).
- **Block period –** (optional) for the administrative blocking, if one is being set.

Unlike the current blocking status, a blocking scheduled for the future is not manifested at all in the account properties. To look for it, check the corresponding report (see ***Report on blockings*** *on page* ***77****).*

- **Unlimited mode –** (must be used with care) turns all charges off this account to zero, thus effectively making all services free.

Tariff link

Tariff link is an object connecting a tariff plan to an account. Tariff links are created, modified or deleted via the user details window (see **Tariffication: Tariff links** on page **47**). Prior to tariff link creation, an account has to be selected to associate the link with.



Tariff link

Tariff link details

Current tariff plan: test tariff 2 (2)

Next tariff plan: test tariff 1 (1)

Accounting period: 17.04.2014 - 17.05.2014 (3)

Tariff link is characterized by the following parameters:

- **Current tariff plan –** is the tariff plan that is acting now.
- **Next tariff plan –** is the tariff plan to switch to by the end of accounting period. May be selected at any time during the current period. Must be compatible with the current plan (see **Tariff plans compatibility** on page **30**). If set to **Do not change**, the tariff link together with all its service links will be prolonged onto the next period. If set to **Disable TP** (this can be done by editing, but not during the creation of the tariff link), the tariff plan will be disconnected and all its service links lost.
- **Change now –** allows to change the tariff plan before the end of the accounting period. The price of the periodic services will be recalculated in that case.
- **Accounting period –** is the period of validity of the current tariff plan. Must be selected from the list of accounting periods (see **Accounting periods** on page **59**) by pressing **Select**. At the end of the period the tariff plan switch occurs.

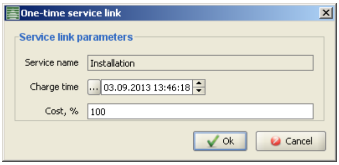
After setting the necessary parameters and pressing ✓ Ok the system will prompt to create the service links for the services having **Attach by default** checked. Other services must be added later by pressing **Add service link** in the tariff link details window.

Service link

Service link is an object connecting a service to an account. Service links are created, modified or deleted via the user details window (see **Tariffication: Service links** on page **47**). Prior to service link creation, an account has to be selected to associate the link with.

Service link parameters are listed below. The actual set of parameters vary depending on the service type.

One-time service link



Service name: Installation
Charge time: 03.09.2013 13:46:18
Cost, %: 100

Service link of this type has the following parameters:

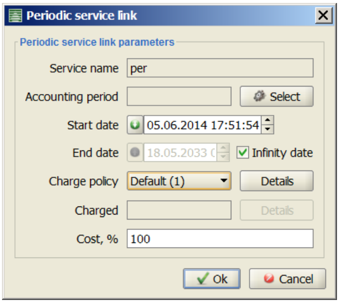
- **Name –** of the service (read-only parameter).
- **Charge time –** at which the charge-off for the service is going to be made. If the date is set in the future, the charge-off will be made exactly then. Otherwise, i.e. if the date is set in the past, the charge-off will be made right after the creation of the service link. The date is interpreted according to the local time of the computer on which the administrator’s interface is running.
- **Cost, % –** is the actual cost of the service for this particular user. It is specified in percents relative to the base cost of the service, and may be altered arbitrarily.

  After charge-off the one-time service link is removed.

Parameters of the service itself are described in **One-time service** on page **62**.

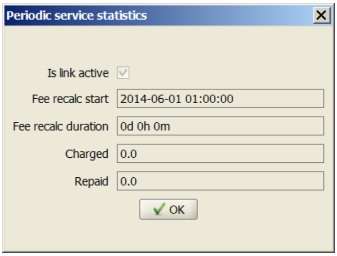
Periodic service link

Service link of this type has the following parameters:



Service name: per
Accounting period:
Start date: 05.06.2014 17:51:54
End date: 18.05.2033 (☑ Infinity date
Charge policy: Default (1)
Charged:
Cost, %: 100

- **Name –** of the service (read-only parameter).
- **Accounting period –** to which the service link refers.
- **Start date –** since when the service is provided to the user.
- **End date –** when the service ceases to be provided and the service link is removed.
- **Charge policy. –** One can select a charge policy to apply to the service link. Press Details to see the chosen charge policy parameters (for more information about the charge policy see **Charge policy** on page **35**).
- **Charged –** the amount of all charges applied to user's account during the current accounting period. Press Details to see the details.
- **Cost, % –** is the actual cost of the service for this particular user. It is specified in percents relative to the base cost of the service, and may be altered arbitrarily. This affects only the periodic component of the service cost.



Is link active ☑
Fee recalc start: 2014-06-01 01:00:00
Fee recalc duration: 0d 0h 0m
Charged: 0.0
Repaid: 0.0

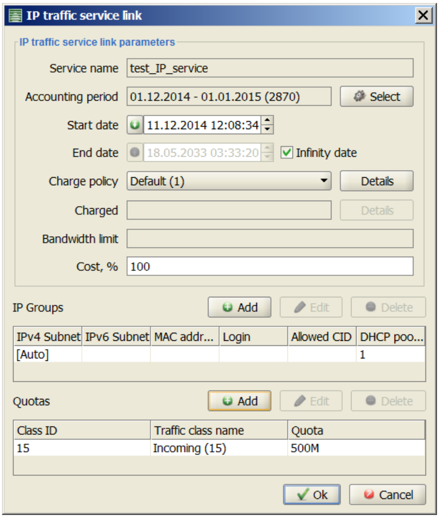
Correction is made at the end of a service period and it uses the value of the filed by the end of the period

Parameters of the service itself are described in **Periodic service** on page **62**.

IP traffic service link

Service link of this type has the same parameters as the periodic service link (see **Periodic service link** on page **110**) plus the following specific parameters:

- **Decrease prepaid traffic on creation –** flag enables the recalculation of prepaid traffic volume granted in the starting period according to the actual portion of the period during which the service has been provided, in a manner similar to the recalculation of periodic fee (see above).
- **Decrease prepaid traffic –** enables similar recalculation in subsequent accounting periods to consider the time spent in blocking.



IP traffic service link

IP traffic service link parameters

Service name: test_IP_service
Accounting period: 01.12.2014 - 01.01.2015 (2870)
Start date: 11.12.2014 12:08:34
End date: 18.05.2033 03:33:20 ☑ Infinity date
Charge policy: Default (1)
Charged:
Bandwidth limit:
Cost, %: 100

IP Groups

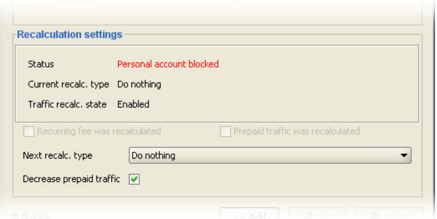
IPv4 Subnet	IPv6 Subnet	MAC addr...	Login	Allowed CID	DHCP poo...
[Auto]					1

Quotas

Class ID	Traffic class name	Quota
15	Incoming (15)	500M

When upgrading from earlier version of UTM (5.2.1-008 or earlier), the Decrease prepaid traffic flag from the account properties (see ***User account*** *on page* ***107****) is transferred here.*

When editing an existing IP traffic service link, the Decrease prepaid traffic on creation flag is absent, and a disabled flag **Prepaid traffic was recalculated** is added instead to show whether a recalculation has already happened in the current period. If the account is blocked, the current type of recalculation is also displayed. The precise moment when the new (edited) recalculation type is to be applied depends on the options:



Recalculation settings

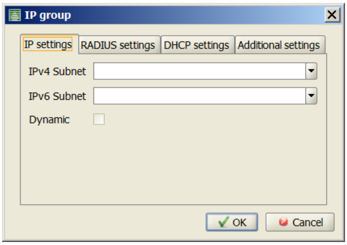
Status: Personal account blocked
Current recalc. type: Do nothing
Traffic recalc. state: Enabled
☐ Recurring fee was recalculated ☐ Prepaid traffic was recalculated
Next recalc. type: Do nothing
Decrease prepaid traffic ☑

  - If a recalculation has already happened, it is at the end of the current period;
  - If no recalculation took place yet and the account is blocked, it is at the end of the current blocking;
  - If no recalculation took place yet and the account is not blocked, it is immediately.

- **IP groups –** that define the user's networks. Traffic identification for further evaluation is based on the IP group parameters together with NetFlow records.

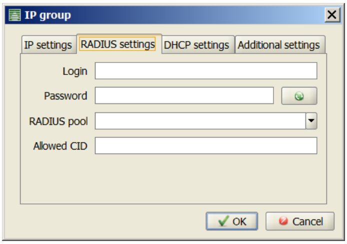
  If the user is linked with a building, the first available address from the zone associated with this building stands for network address.

  An IP group has the following parameters:



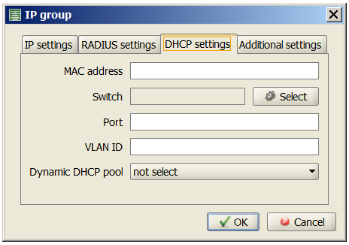
- **IP settings** tab:
  - **IP –** is the IP address and mask of the group’s network. See **IP addresses** on page **40** for the formatting details. This parameter is cached by UTM5 RADIUS.

    The Dynamic flag will be set if the IP group is temporary and has been created automatically for an IP address, assigned by the DHCP server



- **RADIUS settings** tab:
  - **Login and Password. –** Upon authorization with these login and password, the user is granted the first free IP address from the network specified in the properties of the IP group. If there are no free addresses left, authorization is denied.

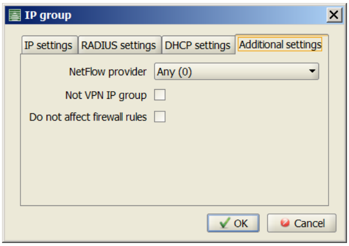
    Pressing ◉ while creating an IP group generates a random password.
    Login is subject to the same requirements as the general UTM5 user login (see **Adding users** on page **45**).
    These parameters are cached by UTM5 RADIUS.
  - **RADIUS pool –** is one of the registered IP pools.
  - **Allowed CID –** is the regular expression against which the value of Calling-Station-ID attribute of the authorization request is checked. If the attribute is not set (not supported by NAS) or does not match the regex, authorization is denied.
    This parameter is cached by UTM5 RADIUS.



Dynamic DHCP pool: not select

- **DHCP settings** tab:
  - **MAC address –** is the parameter used in the firewall rules.
  - **Switch –** is a parameter that lets one link an IP group to a certain switch. UTM5 DHCP uses it for providing IP addresses as a DHCP option 82 parameter.
  - **Port –** is a parameter that lets one link an IP group to a certain switch port. It is also used as a DHCP option 82 parameter for providing IP addresses.
  - **VLAN ID –** is an ID for DHCP client’s VLAN.
  - **Dynamic DHCP pool –** is a pool that will be used to provide an IP address and network settings to the DHCP client. This parameter is mandatory if a static IP address is not set up.

- **Additional settings** tab:
  - **NetFlow provider –** is the firewall providing NetFlow data. If this parameter is set, then only the traffic data coming from this provider will be associated with this IP group.
  - **Not VPN IP group –** is a marker of membership in a non-VPN group.
  - **Do not affect firewall rules –** forbids to apply the firewall rules for this group.



NetFlow provider: Any (0)
Not VPN IP group ☐
Do not affect firewall rules ☐

- **Quotas –** that determine upper limit of traffic to consume. Once the limit is exceeded, the user gets blocked till the end of period. Quotas may be set for different traffic classes separately. In this case each of them is equally capable of triggering the user's blocking.

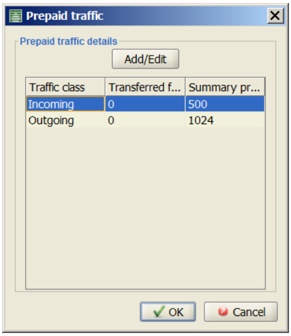
If the end date of an accounting period gets changed while a user stays blocked by quota, the end date of blocking does not change with it.

Parameters of the service itself are described in **IP traffic service** on page **64**.

After adding an IP traffic service link, one can set up additional RADIUS parameters on the **Service links** page. Choose a service link which you want to set up the additional RADIUS parameters for and press the Set RADIUS parameters button. A additional RADIUS parameters window will open (for a more detailed description see **Additional RADIUS attributes** on page **88**). One can also add additional attributes to **Dialup** and **Telephony** service links.

Press ☆ Prepaid traffic to add prepaid traffic. In the popup window choose a traffic class and press Add/Edit. Enter the desired amount of traffic and press ✓ Ok.

Adding prepaid traffic is only possible when the prepaid traffic has not been consumed yet. When the prepaid traffic is consumed, one cannot add prepaid traffic any more.



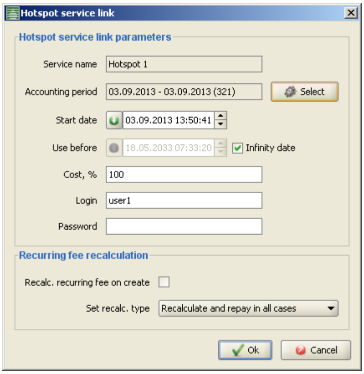
Traffic class	Transferred f...	Summary pr...
Incoming	0	500
Outgoing	0	1024

Hotspot service link

Service link of this type has the same parameters as the periodic service link (see **Periodic service link** on page **110**) plus the following specific parameters:

- **Login, Password –** are to be used for the user’s authorization on the access server.

Hotspot login may not start with a digit.

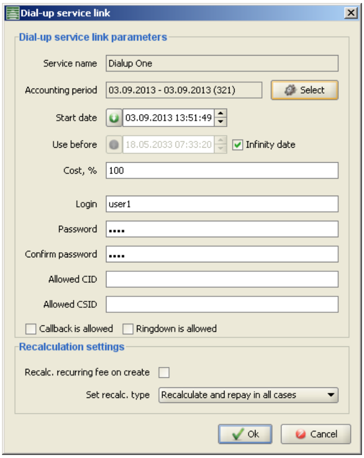


Service name: Hotspot 1
Accounting period: 03.09.2013 - 03.09.2013 (321)
Start date: 03.09.2013 13:50:41
Use before: 18.05.2033 07:33:20 ☑ Infinity date
Cost, %: 100
Login: user1
Password:
Recalc. recurring fee on create ☐
Set recalc. type: Recalculate and repay in all cases

Parameters of the service itself are described in **Hotspot service** on page **65**.

Dialup service link

Service link of this type has the same parameters as the periodic service link (see **Periodic service link** on page **110**) plus the following specific parameters:



Service name: Dialup One
Accounting period: 03.09.2013 - 03.09.2013 (321)
Start date: 03.09.2013 13:51:49
Use before: 18.05.2033 07:33:20 ☑ Infinity date
Cost, %: 100
Login: user1
Password: ••••
Confirm password: ••••
Allowed CID:
Allowed CSID:
☐ Callback is allowed ☐ Ringdown is allowed
Recalc. recurring fee on create ☐
Set recalc. type: Recalculate and repay in all cases

- **Login, Password –** are to be used for the user’s authorization on the access server. On successful authorization the server establishes connection and issues the dynamic IP from a pool. These parameters are cached by UTM5 RADIUS.
  While accepting the login, UTM5 RADIUS may automatically cut off the prefix defined by the `radius_realm` parameter (see **Interface parameters** on page **152**).
- **Allowed CID –** is a regular expression to check against the Calling-Station-ID attribute of the authentication request. This parameter is cached by UTM5 RADIUS. If the regex is set, but the attribute does not match it or is missing, the authorization is denied.
- **Allowed CSID –** is a regular expression to check against the Called-Station-ID attribute of the authentication request. This parameter is cached by UTM5 RADIUS. If the regex is set, but the attribute does not match it or is missing, the authorization is denied.
- **Callback allowed –** instructs the RADIUS server to check the incoming login (the part after colon) against the Callback standard. This parameter is cached by UTM5 RADIUS.

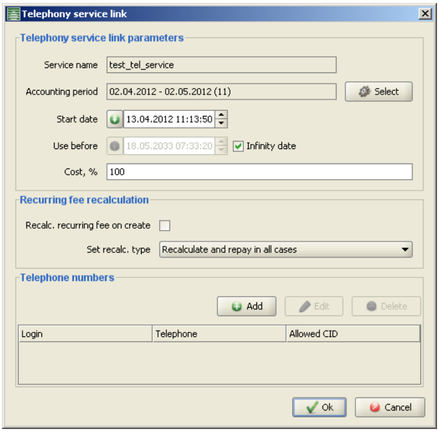
- **Ringdown allowed –** instructs the RADIUS server to check the incoming login for all non-Callback calls. This parameter is cached by UTM5 RADIUS. If neither **Callback allowed** nor **Ringdown allowed** are set, the authorization with the given login and password is denied.

  Parameters of the service itself are described in **Dialup service** on page **66**.

Telephony service link

Service link of this type has the same parameters as the periodic service link (see **Periodic service link** on page **110**) plus one specific parameter:

- **Telephone numbers –** is the table with phone numbers, containing the following information on each entry:
  - **Login –** to be used for identification of the service link.
  - **Incoming trunk, Outgoing trunk, PBX ID –** against which every call must be checked (if set).
  - **Telephone –** number issued to the user upon registration (if set).
  - **Password –** to be used for the user’s registration or a call authorization.
  - **Allowed CID –** which is a regular expression to check against the Calling-Station-ID attribute of the authentication request. If the regex is set, but the attribute does not match it or is missing, the authorization is denied.



Telephony service link

Telephony service link parameters

Service name: test_tel_service

Accounting period: 02.04.2012 - 02.05.2012 (11)

Start date: 13.04.2012 11:13:50

Use before: 18.05.2033 07:33:20 ☑ Infinity date

Cost, %: 100

Recurring fee recalculation

Recalc. recurring fee on create ☐

Set recalc. type: Recalculate and repay in all cases

Telephone numbers

Login	Telephone	Allowed CID

The set of parameters identifying the telephony service link (which are: login, incoming trunk, outgoing trunk, and PBX ID) must be unique. In each set at least one of these parameters must be filled, i.e. non-empty. If a call matches several service links by some parameters, the one with the greatest number of matches is selected.

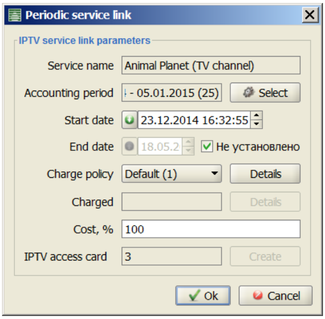
Parameters of the service itself are described in **Telephony service** on page **67**.

IPTV service link

Service link of this type has the same parameters as the periodic service link (see **Periodic service link** on page **110**) plus one specific parameter:

- **IPTV access card –** is the user's IPTV access card number. If this card was not created earlier, one can create it in this window. Press Create to create a new access card.

  For more information on access card, see the IPTV administrator's manual.



Periodic service link

IPTV service link parameters

Service name: Animal Planet (TV channel)

Accounting period: - 05.01.2015 (25)

Start date: 23.12.2014 16:32:55

End date: 18.05.2 ☑ Не установлено

Charge policy: Default (1)

Charged:

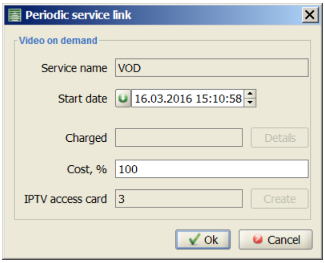
Cost, %: 100

IPTV access card: 3

Video on demand service link

This service link has the following parameters:

- **Start date –** is the date and time when service starts to be provided
- **Cost, % –** is the actual cost of the service for this particular user. It is specified in percents relative to the base cost of the service, and may be altered arbitrarily. This affects only the periodic component of the service cost.



Periodic service link

Video on demand

Service name: VOD

Start date: 16.03.2016 15:10:58

Charged:

Cost, %: 100

IPTV access card: 3

Correction is made at the end of a service period and it uses the value of the filed by the end of the period

- **IPTV access card –** is the user's IPTV access card number. If this card was not created earlier, one can create it in this window. Press Create to create a new access card.

  For more information on access card, see the IPTV administrator's manual.

Tray icon

When the control center is working, it is represented by an icon ▤ in the system tray. The icon has a context menu of its own, which contains the following items:

- **Main window –** activates the main window of the program.
- **Frames – select one to move on top –** (active when there are multiple open windows) selects and activates any of the control center windows, except the main one.

- **Processes – select one to cancel –** (active when there are processes running; at that, the icon changes to) selects and terminates a process. May be worthwhile if the process is taking longer than expected.
- **Cancel printing –** (active when there is a printing task running) terminates printing.
- **Exit –** stops the control center.

INSTALLATION AND INITIAL SETUP

In Brief

Installation and launch

Prior to installation make sure that the computer has valid system date and time set. Otherwise NetUP UTM5 may work incorrectly.

1. Install the billing system on a server according to the instructions below. Launch `utm5_core` program.
2. Initialize the database.
3. Launch the UTM5 core and (if necessary) RADIUS server and UTM5 RFW affector.
4. Launch the `UTM_admin` program (requires Java; see **Installation and startup** on page **129**).
5. Change passwords for the system user **init**.

Besides the installed program files, UTM5 requires disk space for the files with raw traffic data (their size depend on the system load and may achieve significant values), as well as for log files.

Tariffication system setup

After the first installation of the system:

1. Create appropriate accounting periods in the administrator interface.
2. Create appropriate traffic classes and time ranges in the tariffication system interface.
3. Specify currency exchange rates in the system. By default, exchange rate of euro to the internal unit is 1.0, i.e. users' balance is shown in euro. If you want to maintain personal accounts in other currency, set its exchange rate to 1 and alter the other rates accordingly.
4. Create services, specify their costs and terms of validity.
5. Start adding users.

System testing

1. Set up and launch a NetFlow probe.
2. Having downloaded some amount of traffic towards a client, check whether the traffic data are represented accurately in the detailed traffic report and common traffic report.

64-bit distributives of UTM5 are available for Debian Squeeze and FreeBSD 9. Consistent operation of a 32-bit version of UTM5 on a 64-bit OS in compatibility mode is not guaranteed.

Installation: Detailed

Linux with rpm

Linux (RHEL5, CentOS5, or other rpm-based library-compatible distributive) should be installed on the server together with additional packages: Apache 2 web server supporting SSL and MySQL 5.0.x or PostgreSQL 8.x database server. We recommend to use MySQL with InnoDB support to ensure maximum reliability and integrity of stored data.

The following libraries are also required:

- `openssl`;
- `glibc`;
- `libxml2`;
- depending on the particular OS version:

For RHEL5:	**For CentOS5:**
`krb5-libs libgssapi keyutils-libs libsepol libselinux e2fsprogs-libs`	`zlib libgcc e2fsprogs-libs libgssapi krb5-devel krb5-libs keyutils-libs-devel libsepol libselinux`

UTM5 system is incompatible with the SELinux extension.

1. To start the installation, run:

```
rpm -ihv --nodeps utm5-2.1.xxx.rpm
```

The directory `/netup` will be created to store the basic work files, configuration files, and log files.

Launch scripts will be also copied:

- `/etc/init.d/utm5_core`
- `/etc/init.d/utm5_radius`
- `/etc/init.d/utm5_rfw`

1. Create the database (see **Database creation** on page **124**).
2. Import the license key into the database (see **License key activation** on page **125**).
3. Create indexes (see **Index creation** on page **126**).
4. If all previous commands have been run normally, start the billing system core as follows:

```
/etc/init.d/utm5_core start
```

5. To start the UTM5 core automatically at the Linux OS startup, run the following command:

```
chkconfig --add utm5_core
chkconfig utm5_core on
```

and similar commands for `utm5_radius` and `utm5_rfw`, if needed.

Linux Debian

Linux Debian (Lenny or Squeeze) should be installed on the server together with additional packages: Apache 2 web server supporting SSL and MySQL 5.0.x or Postgr-eSQL 8.x database server. We recommend to use MySQL with InnoDB support to ensure maximum reliability and integrity of stored data.

The following libraries are also required:

- `openssl`;
- `glibc`;
- `e2fsprogs`;
- `libkrb53`;
- `libxml2`;
- depending on the particular OS version:

For Debian Squeeze:	**For Debian Wheezy:**
`zlib libldap2 libtasn1-3 libgpg-error0 libgnutls26 libgcrypt11 libsasl2-2 libgssapi-krb5-2 libkeyutils1`	`zlib-bin libldap-2.4-2 libtasn1-3 libgpg-error0 libgnutls26 libgcrypt11 libsasl2-2 libgssapi-krb5-2 libkeyutils1`

⚠ *UTM5 system is incompatible with the SELinux extension.*

1. To start the installation, run:

```
dpkg -ih utm5-2.1.xxx.tgz
```

The directory `/netup` will be created to store the basic work files, configuration files, and log files.

Launch scripts will be also copied:

  - `/etc/init.d/utm5_core`
  - `/etc/init.d/utm5_radius`
  - `/etc/init.d/utm5_rfw`

1. Create the database (see **Database creation** on page **124**).
2. Import the license key into the database (see **License key activation** on page **125**).
3. Create indexes (see **Index creation** on page **126**).

4. If all previous commands have been run normally, start the billing system core as follows:

```
/etc/init.d/utm5_core start
```

5. To start the UTM5 core automatically at the OS startup, run the following command:

```
update-rc.d utm5_core defaults
```

   and similar commands for `utm5_radius` and `utm5_rfw`, if needed.

FreeBSD

FreeBSD OS 9.x, or 10.x should be installed on the server with additional packages: Apache 2 web server supporting SSL and MySQL 5.6.x or PostgreSQL 8.x database server. We recommend to use MySQL with InnoDB support to ensure maximum reliability and integrity of stored data.

Also, the following libraries are required:

- `openssl`;
- `libxml2`.

1. To start the installation, run

```
pkg add utm5-3.xxx.tbz
```

   The directory `/netup` will be created to store the basic work files, configuration files, and log files.

   Launch scripts will be also copied:

   - `/usr/local/etc/rc.d/utm5_core.sh`
   - `/usr/local/etc/rc.d/utm5_radius.sh`
   - `/usr/local/etc/rc.d/utm5_rfw.sh`

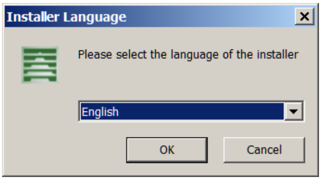
1. Create the database (see **Database creation** on page **124**).
2. Import the license key into the database (see **License key activation** on page **125**).
3. Create indexes (see **Index creation** on page **126**).
4. If all previous commands have been run normally, start the billing system core as follows:

```
/usr/local/etc/rc.d/utm5_core.sh start
```

Windows

When updating from an earlier UTM5 version, Apache and MySQL should be already installed. If UTM5 is being installed for the first time, you should install MySQL and Apache web server before installing UTM5.

1. Launch the UTM5 installation wizard `utm5-3.x.xxx.exe`. The language selection window will appear.

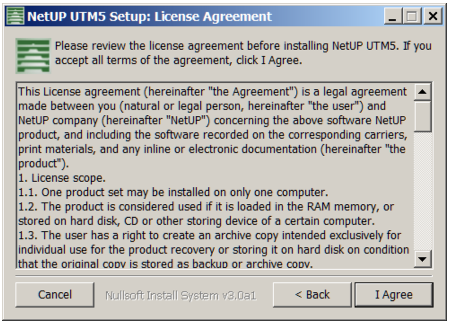


Installer Language

Please select the language of the installer

English

2. Select **English** and press **OK**. The license agreement window will appear. Press **I Agree**.

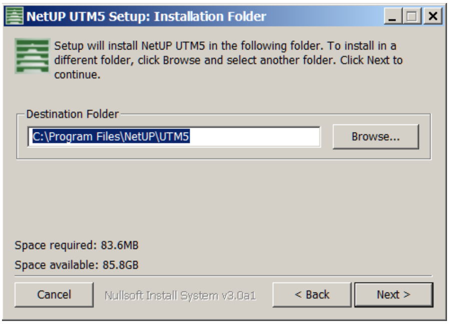


NetUP UTM5 Setup: License Agreement

Please review the license agreement before installing NetUP UTM5. If you accept all terms of the agreement, click I Agree.

This License agreement (hereinafter "the Agreement") is a legal agreement made between you (natural or legal person, hereinafter "the user") and NetUP company (hereinafter "NetUP") concerning the above software NetUP product, and including the software recorded on the corresponding carriers, print materials, and any inline or electronic documentation (hereinafter "the product").
1. License scope.
1.1. One product set may be installed on only one computer.
1.2. The product is considered used if it is loaded in the RAM memory, or stored on hard disk, CD or other storing device of a certain computer.
1.3. The user has a right to create an archive copy intended exclusively for individual use for the product recovery or storing it on hard disk on condition that the original copy is stored as backup or archive copy.

3. The installation path selection window will appear. Press **Browse** if you wish to change the default path. Then press **Next**.



NetUP UTM5 Setup: Installation Folder

Setup will install NetUP UTM5 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

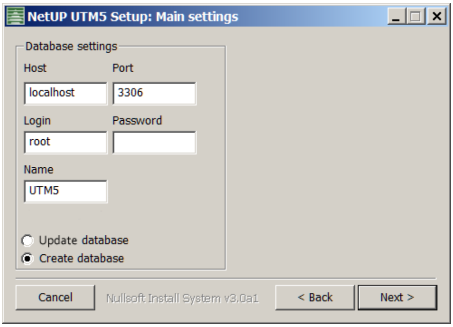
Destination Folder: C:\Program Files\NetUP\UTM5

Space required: 83.6MB

Space available: 85.8GB

4. The MySQL settings window will appear. On first installation check **Create database**. On update, check **Update database.** Then press **Next**.
5. In the next window press **Browse**, select the path to the license file `reg.sql` and press **Next**.
6. The installation program will proceed.

The UTM5 billing system has been installed.



NetUP UTM5 Setup: Main settings

Database settings

Host: localhost

Port: 3306

Login: root

Password:

Name: UTM5

○ Update database

● Create database

*For proper functioning of UTM5 billing make sure that your system settings (**Regional and Language Options**) specify the "."symbol as the fractional part delimiter.*

The UTM5 server is installed as a Windows NT service named `utm5_core`. In order to start it run the following command:

```
net start utm5_core
```

In order to run UTM5 server in debug mode (all information is output to the screen instead of log files) select **Start** – **Programs** – **UserTrafManager 5.0** – **UTM5 Core Debug Mode**, or run the following command:

```
C:\program files\NetUP\UTM5\bin\utm5_core.exe -d
```

To install or remove the `utm5_core` service, run `utm5_core.exe` with the options `--install` or `--uninstall`, correspondingly:

```
C:\Program Files\NetUP\UTM5\bin\utm5_core.exe --install
Successfully created utm5_core service
```

```
C:\Program Files\NetUP\UTM5\bin\utm5_core.exe --uninstall
Successfully deleted utm5_core service
```

Auxiliary operations

Database creation

On UNIX systems you have to create the database manually and feed into SQL the commands that create the tables and populate them with some predefined values. The necessary commands are listed in the files UTM5_MYSQL.sql (for MySQL), UTM5_PG.sql (for PostgreSQL), and an additional file UTM5_en.sql (for both). Normally this is done as follows.

- For MySQL (UTF-8 encoding should be specified on database creation, unless set as default):

```
mysql -e "create database UTM5 DEFAULT CHARACTER SET=utf8;"
mysql UTM5 < /netup/utm5/UTM5_MYSQL.sql
mysql UTM5 < /netup/utm5/UTM5_en.sql
```

- For PostgreSQL:

```
createdb -U postgres UTM5
psql -f /netup/utm5/UTM5_PG.sql -U postgres UTM5
psql -f /netup/utm5/UTM5_en.sql -U postgres UTM5
```

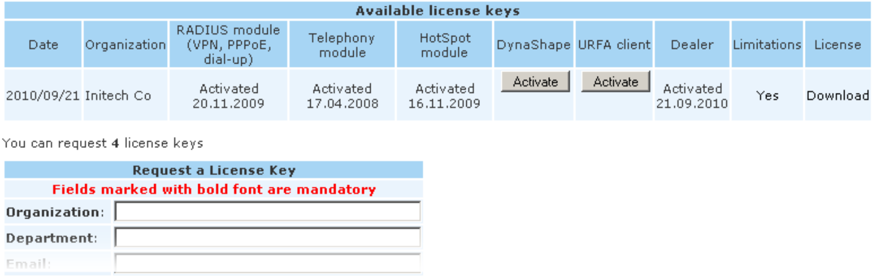
On Windows the installation package creates databases automatically.

If the database name is different from the standard value (UTM5, as in the examples above), it has to be changed in the UTM5 config file (the `database` parameter, see **UTM5 core: Config file** on page **147**), and also in other SQL commands below.

Database encoding is also mentioned in the `database_charset` parameter of the UTM5 config file. If an alternative encoding is used, this parameter's value should be changed accordingly.

License key creation

The license key request form is located at the client's personal cabinet on **https://www.utm-billing.com/customer.php** under **License Keys** and is activated upon purchasing the license. Once the request is made, the key itself with an activation form appears on the same page. Each of the available modules is marked with the **Activate** button which, once pressed, turns into the **Activated** mark. Activate all modules you need and press **Download** to download the key (i.e. the `reg.sql` file).



Available license keys

Date	Organization	RADIUS module (VPN, PPPoE, dial-up)	Telephony module	HotSpot module	DynaShape	URFA client	Dealer	Limitations	License
2010/09/21	Initech Co	Activated 20.11.2009	Activated 17.04.2008	Activated 16.11.2009	Activate	Activate	Activated 21.09.2010	Yes	Download

You can request **4** license keys

Request a License Key

Fields marked with bold font are mandatory

Organization:

Department:

Email:

When some new modules are purchased, the key has to be activated and downloaded anew.

License key activation

For generating and downloading the license key, see ()**License key creation** on page **125**.

In order to activate the license key on Unix systems, feed into SQL the commands contained in the `reg.sql` file.

While executing the commands from `reg.sql`, non-critical errors may pop up indicating that you are trying to create an entity that already exists or to delete something non-existing; they may be ignored.

On Windows the installation package automatically prompts the user for the path to the `reg.sql` file. If this was not done, execute

```
[path to MySQL]mysql.exe UTM5 < reg.sql
```

By default the MySQL executable files are located at `C:\Program Files\MySQL\MySQL Server 5.0\bin\`.

Index creation

In order to create database indexes on Unix systems, feed into SQL the commands contained in the UTM5_indexes.sql file, which is normally done as follows.

- For MySQL:

```
mysql -f UTM5 < /netup/utm5/UTM5_indexes.sql
```

- For PostgreSQL:

```
psql -f /netup/utm5/UTM5_indexes.sql -U postgres UTM5
```

While creating the indexes, non-critical errors may pop up indicating that you are trying to delete an already deleted entity; they may be ignored.

On Windows the installation package creates the indexes automatically.

Update

1. Stop all UTM5 components.
2. Make backup copies of the config files.
3. Make backup of the SQL database.
4. Remove old version of UTM5.
5. Install new version of UTM5.

During the automatic update of the database structure, non-critical errors like ERROR 1060 (column already created) or ERROR 1091 (column already deleted) may pop up; they may be ignored.

6. Restore config files from backup.

7. Perform additional activities specific to the particular update, if required by the instructions.

When upgrading from any version before UTM5.3-001 to this or later version, you have to convert the telephony-related database tables with the utility located at `/netup/utm5/bin/tel_conv` *(answer "Yes" when prompted). This utility accepts the following command line keys:*

Key	**Meaning**
`-c <path>`	Path to the UTM5 config file
`-d`	Remove all data from the new tables (relevant for repeated runs)
`-f`	Convert without checking (otherwise the program would check the data for consistency and stumble upon any conflict, in which case it would need to be run again after manual resolution of the conflict)

8. Start the necessary components of UTM5.

All components must have the same assembly number. Components from different assemblies can not be used simultaneously.

USAGE EXAMPLES

8

Introduction

This chapter contains typical scenarios of UTM usage. All actions are performed via the interfaces of the control center which itself is described in **Installation and startup**. Complete description of the control center interface pages is given in **Administrator's interface** on page **41**.

To perform the initial setup of the UTM system one generally needs to do the following:

- Create accounting periods (see **Creating accounting periods** on page **130**);
- Create traffic classes (see **Creating traffic classes** on page **131**);
- Create services (see **Creating services** on page **132**);
- Create users (see **Creating users** on page **133**);
- Assign services to users (see **Creating service links** on page **136** and **Creating tariff links** on page **138**).

Installation and startup

To install and start the control center:

1. Download the administrator's interface located in the client's personal cabinet on **https://www.utm-billing.com/customer.php** under **Downloads**. The file is called `utm_admin.zip`.
2. Unpack the archive on the administrator's workstation (i.e. the computer which will be used to control the UTM5 system).

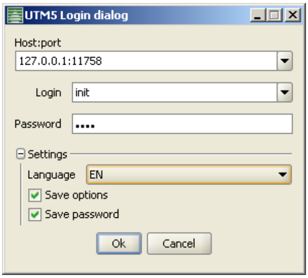
Java Runtime Environment (JRE) version 8.0 (Java 1.8.x) or above is required in order to use the control center
JRE distributive is available for free at **http://j-ava.com**.

3. Start the control center either by clicking on the file `UTM_admin.jar` or from the command line by executing

```
java -jar UTM_admin.jar
```

The login dialog window will appear.

4. Enter the IP address and colon-separated port number to connect to. If the port number is omitted, the default value of 11758 is assumed.
5. Enter the login and password. By default, login is **init** and password is **init**.



UTM5 Login dialog

Host:port 127.0.0.1:11758

Login init

Password ••••

Settings

Language EN

☑ Save options

☑ Save password

6. In the **Settings** group of parameters select the language to use.

Note that the selected language is not applied immediately to the login dialog itself. Instead, the language switch occurs on the next launch of the program.

7. Check **Save options** if you want to save the parameters just entered (except for password) in the settings file for use during subsequent launches. Check **Save password** if you also want to save the password as well.

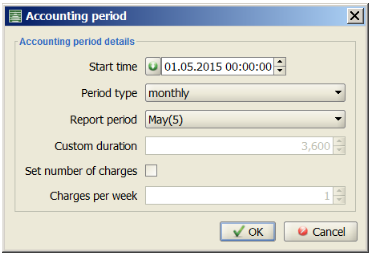
It is highly recommended to change the password for the system user ***init*** *immediately after logging in for the first time (see Administrator’s interface: Change password on page 92).*

Creating accounting periods

Accounting period (see **Basic system objects: Accounting periods** on page **29**) is a period of time to which various periodic activities, including charge-offs, are related.

To create an accounting period:

1. Click **Accounting periods** on the left pane under **Tariffication**. The list of available accounting periods will appear.



Start time 01.05.2015 00:00:00
Period type monthly
Report period May(5)
Custom duration 3,600
Set number of charges ☐
Charges per week 1

2. Press Add to create a new accounting period. An **Accounting period** window will show up.
3. Select the starting date of the period.
4. Select the type of the period (daily, weekly, monthly, quarterly, annual, or custom; in the last case enter also the duration).
5. Press Ok to finalize the creation of the new accounting period.

   When the period finishes, a new one of the same type is created automatically.

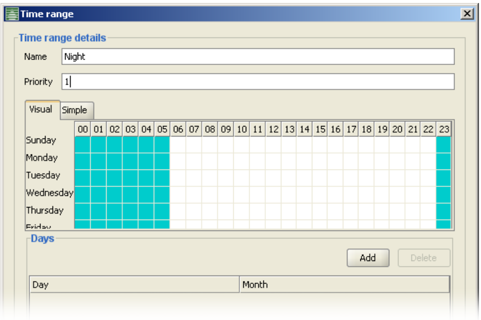
Creating time ranges

Time range (see **Basic system objects: Time ranges** on page **37**) is a period of time, or a set of such periods, used by the system to define time-dependent behavior.

To create a time range:

1. Click **Time ranges** on the left pane under **Tariffication**. The list of available time ranges will appear.
2. Press Add to create a new time range. A **Time range** window will show up.
3. Set the name of the new time range (e.g **Night**).

4. Set the new range's priority to 1 (the higher the number, the higher the priority).
5. In the visual editor select the night hours.



Name: Night

Priority: 1

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sunday	■	■	■	■	■	■																		■
Monday	■	■	■	■	■	■																		■
Tuesday	■	■	■	■	■	■																		■
Wednesday	■	■	■	■	■	■																		■
Thursday	■	■	■	■	■	■																		■

Use Shift in order to select multiple cells at a time. E.g. to select the whole time range select 00 hours on Sunday, press Shift and select 23 hours on Saturday.

6. Press ✓ Ok to finalize the creation of the new time range.

Creating traffic classes

To classify traffic (see **Basic system objects: Traffic classes** on page **28**), the UTM5 system contains two predefined classes, namely **Incoming** and **Outgoing**. The **Incoming** class has ID set to 10 and consists of a single subclass with its **Destination** parameter set to the address and mask of the local network. The **Outgoing** class has ID set to 20 and a single subclass with local address for **Source**. You may want to create additional classes, say, to charge different prices for the traffic at different times of day.

*In order to create traffic classes with time-dependent condition of membership, one has to create the corresponding time ranges beforehand, see **Creating time ranges** on page **130**.*

Creating new traffic classes:

1. Press **Traffic Classes** on the left pane under **Tariffication**. The list of existing traffic classes will appear.
2. Press Add to create a new traffic class. The **Traffic class** window will show up.
3. Set the traffic class ID to **15**.
4. Set the traffic class name to **Night Incoming**.
5. Select **Night** in the **Time range** drop-down list.
6. Press **Add** above the list of traffic subclasses. The traffic subclass properties window will show up.

7. In the **Addressee** group enter the IP address and subnet mask for local network, in the **Source** group enter the source network address and subnet mask (e.g. enter 0.0.0.0/0 if source of the traffic doesn't matter) and press [✓ Ok].
8. After creating the subclass press [✓ Ok] in the **Traffic class** window to finalize the creation of the new traffic class.
9. In a similar manner create the class **Day Outgoing** with the following properties:
    - ID: **25**;
    - Time range: **Day** (supposed you've already created this time range);
    - Subclass properties: enter local IP address/mask into the **Source** group.
1. In a similar manner create the class **Internal** with the following properties:
    - ID: **1000**;
    - Time range: leave default value (**Not defined**);
    - Subclass properties: enter local IP address/mask into both the **Source** and **Destination** groups.

Creating services

UTM5 may contain services of various types (see **Basic system objects: Services** on page **31**) which in turn may require some type-dependent prerequisites. In particular, to create an IP traffic service, one has to create the necessary traffic classes in the first place (see **Creating traffic classes** on page **131**).

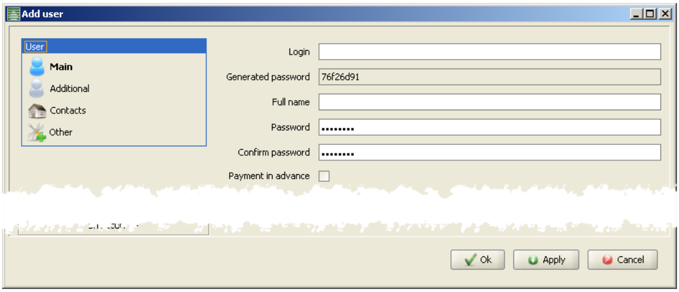
To create a new IP traffic service:

1. Click **Services** ★ on the left pane under **Tariffication**. The list of existing services will appear.
2. Press [⊕ Add] to create a new service. The **Service** window will appear.
3. Enter the service name.
4. Set the service type to **IP traffic**.
5. A set of additional shortcuts will appear on the left pane.
6. In **Service parameters** enter the periodic fee and select a charge-off method (at the beginning, or at the end of an accounting period, or flow method).
7. In **Tariffication borders** press [⊕ Add] above the list of borders.
8. In the window that appears, select a traffic class for the border, enter the border position in bytes (0) and cost for traffic exceeding the border in currency units per megabyte. Press [✓ Ok].
9. In order to add prepaid traffic to the service, press [⊕ Add] above the list of prepaid units in **Prepaid traffic**.

10. In the window that appears select the class of prepaid traffic and enter its volume in bytes. Press ✓ Ok.
11. Press ✓ Ok in the Service window to finalize the creation of the new service.

Creating users

New user account is created via the dialog window for adding users (called by pressing the button in the list of users). Required information is a user login and a password. On creation of a new user account a password is being generated automatically, yet it may be changed. A personal account is being created along with the user account.



Generated password: 76f26d91

To create a new user:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Press Add. The user properties window will show up.
3. In the user properties window enter the user's login and (if necessary) personal information.
4. Press **Apply**. A set of additional shortcuts will appear on the left pane.
5. Select **Other** on the left pane. In the list **Currency** select the currency for transactions with the user.
6. Switch to **Tariff links** under **Tariffication** and press Add.
7. From the pull-down lists select a tariff plan for the current accounting period and also one for the next period.
8. Press **Select** next to the **Accounting period** field, select an appropriate accounting period from the appeared list and press ✓ Ok.
9. Switch to **Service links** under **Tariffication** and add the service links for the user.
10. Press ✓ Ok in the **Add user** window to finalize the creation of the user.

When a user account has been created, one may start adding services. See **Creating service links** on page **136** and **Creating tariff links** on page **138** for details.

Removing a user

To remove a user:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. If the user's accounts have some service links attached to them:
    - 3.1. On the left pane of the user properties window open **Service links** under **Tariffication**. The list of service links will appear.
    - 3.2. Remove each service link by selecting it, pressing Delete and selecting **OK** in the confirmation window.
    - 3.3. Repeat the previous step with the other user's accounts, if any.
4. If the user's accounts have some tariff links attached to them:
    - 4.1. On the left pane of the user properties window open **Tariff links** under **Tariffication**. The list of tariff links will appear.
    - 4.2. Remove each tariff link by selecting it, pressing Delete and selecting **OK** in the confirmation window.
    - 4.3. Repeat the previous step with the other user's accounts, if any.
5. Press Close to close the user properties window.
6. Select the user in the list and press Delete.

The user can not be removed while still having some service or tariff links attached.

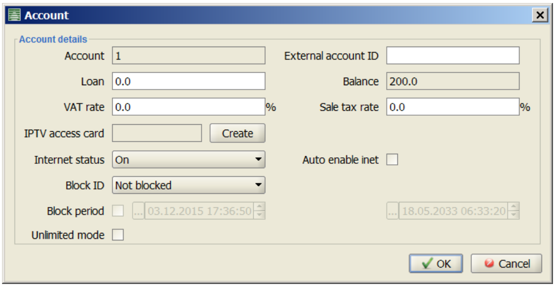
Creating account

The user's primary account is created automatically together with the user. Besides, an arbitrary number of additional accounts may be created afterwards.

To create an additional account:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. On the left pane of the user properties window open **Accounts** under **Tariffication**. The list of user's accounts (initially containing only one account) will appear.

4. Press Add. The account properties window will show up.



Account

Account details

Account	1	External account ID	
Loan	0.0	Balance	200.0
VAT rate	0.0 %	Sale tax rate	0.0 %
IPTV access card	Create		
Internet status	On	Auto enable inet	☐
Block ID	Not blocked		
Block period	☐ 03.12.2015 17:36:50		18.05.2033 06:33:20
Unlimited mode	☐		

5. Select the Internet status for the account being created (**On** / **Off**).
6. Select **Credit** for the account (a sum to be put on this account upon creation).
7. Select **Block ID** for the account (see the list in **Accounts** on page **26**). In case of blocking type other than **No** you may check the **Block period** flag and set the time span for the blocking to persist.
8. Enter the tax rates, namely **VAT rate** and **Sale tax rate**.
9. If necessary, check the flags **Don't charge recurrent fee** and **Decrease prepaid traffic** for the system blocking.

Default values of these properties are determined by the `block_recalc_abon`*,* `block_recalc_prepaid`*, and* `default_vat_rate` *parameters (see* ***Interface parameters*** *on page* ***152****).*

10. If necessary, check the **Unlimited mode** flag.
11. Press Ok to finalize the creation of the new account.

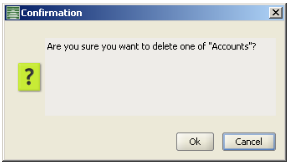
Removing an account

To remove an additional account:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. If the account has some service links attached to it:
    3.1. On the left pane of the user properties window open **Service links** under **Tariffication**. The list of service links will appear.
    3.2. If another account is selected in the drop-down list, select the one you need.
    3.3. Remove each service link by selecting it, pressing Delete and selecting **OK** in the confirmation window.
4. If the account has some tariff links attached to it:
    4.1. On the left pane of the user properties window open **Tariff links** under **Tariffication**. The list of tariff links will appear.
    4.2. If another account is selected in the drop-down list, select the one you need.
    4.3. Remove each tariff link by selecting it, pressing Delete and selecting **OK** in the confirmation window.

5. On the left pane of the user properties window open **Accounts** under **Tariffication**. The list of user's accounts will appear.
6. Select the required account in the list and press Delete.
7. Press **OK** in the confirmation window to finalize the deletion of the account.

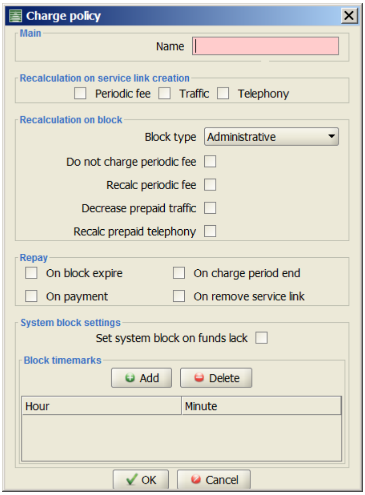
The user's primary account can not be removed.



Are you sure you want to delete one of "Accounts"?

Creating a charge policy

To create a charge policy:



1. Open **Charge policies** page in the **Tariffication** group of pages.
2. Press Add. A charge policy window will open.
3. Enter charge policy's name.
4. Check all flags in the **Recalculation on service link creation group** (for more information on recalculation see **Basic system objects: Charge policy** on page **35**).
5. Check the checkboxes in the **Recalculation on block** group as needed. Switch to another **Block type** to check the checkboxes for all the rest block types. The parameters set for the previous block type are stored and won't be lost after switching. It acts as if it were tabs.
6. In the next group - **Repay**, check the events that you want to be coupled with refund when excessive amount of money was withdrawn from the user's account.
7. Check the **Set system block on funds lack** checkbox in the **System block settings** group.
8. Press Ok to create the charge policy.

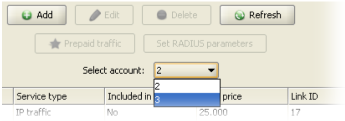
Creating service links

Service link is a system object linking a service to a user's account. Besides the service and account, an accounting period is required to create a new service link (see **Creating accounting periods** on page **130**).

To create a new service link:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.

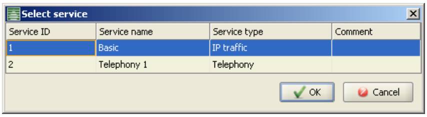
2. Select the user in the list and press Edit. The user properties window will show up.
3. On the left pane of the user properties window open **Service links** under **Tariffication**. The list of service links attached to the primary account will appear.
4. Select another account if necessary. The list of service links will switch to that of the selected account.



Select account: 2

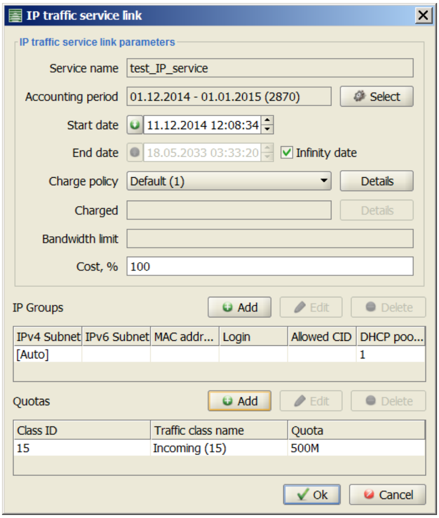
Service type	Included in	price	Link ID
IP traffic	No	25.000	17

5. Press Add. The service selection window will show up.



Service ID	Service name	Service type	Comment
1	Basic	IP traffic	
2	Telephony 1	Telephony	

6. Select the service from the list and press Ok. The service link properties window will show up.
7. Press Select and select the accounting period for the service link.
8. Select the starting date for the service.
9. Select the ending date, or check the **Infinite date** flag.
10. Select a charge policy.
11. Set the service cost correction if necessary.
12. Press Add to add an IP group.



Service name: test_IP_service
Accounting period: 01.12.2014 - 01.01.2015 (2870)
Start date: 11.12.2014 12:08:34
End date: 18.05.2033 03:33:20 ☑ Infinity date
Charge policy: Default (1)
Charged:
Bandwidth limit:
Cost, %: 100

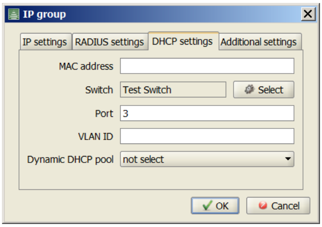
IP Groups

IPv4 Subnet	IPv6 Subnet	MAC addr...	Login	Allowed CID	DHCP poo...
[Auto]					1

Quotas

Class ID	Traffic class name	Quota
15	Incoming (15)	500M

13. In the IP group window go to DHCP settings tab and choose a dynamic DHCP pool which will be used for giving an IP address to user. You may also choose switch and port to which the user is connected.
14. Press Ok to create the IP group.
15. Press Ok to finalize the creation of the new service link.



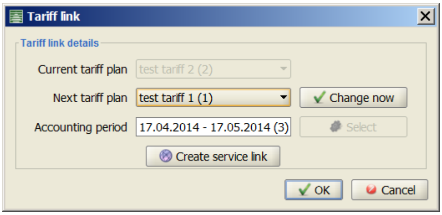
MAC address:
Switch: Test Switch
Port: 3
VLAN ID:
Dynamic DHCP pool: not select

Creating tariff links

Tariff link is a system object linking a tariff plan to a user’s account. Besides the plan and account, an accounting period is required to create a new tariff link (see **Creating accounting periods** on page **130**).

To create a new tariff link:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. On the left pane of the user properties window open **Tariff links** under **Tariffication**. The list of tariff links attached to the primary account will appear.
4. Select another account if necessary. The list of tariff links will switch to that of the selected account.
5. Press Add. The tariff link properties window will show up.
6. Select the **Current tariff plan** from the drop-down list.
7. Select the **Next tariff plan** from the drop-down list, or leave the default choice **Do not change**.



Current tariff plan: test tariff 2 (2)
Next tariff plan: test tariff 1 (1)
Accounting period: 17.04.2014 - 17.05.2014 (3)

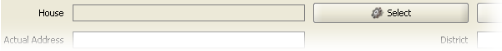
8. Press Select and select the accounting period for the tariff link.
9. Press Ok to finalize the creation of the new tariff link.
10. If the current tariff plan contains some one-time services having their **Attach by default** flag set, each of those will be manifested by a prompt window. Select the date and time of the charge and press Ok.

Linking a user to a house

UTM5 system is capable of maintaining a list of houses (see Reference book: Buildings on the left pane). To link a user to a house:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. On the left pane of the user properties window open **Contacts** under **User**. The user contacts page will appear.

4. Press Select next to **House** and select the building from the list.

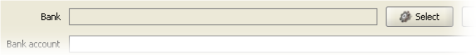


5. Press Ok to close the list of buildings.
6. Press Ok in the user properties window to save the changes.

Linking a user to a bank

UTM5 system is capable of maintaining a list of banks (see Reference book: Banks on the left pane). To link a user to a bank:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up.
3. On the left pane of the user properties window open **Additional** under **User**. The additional user information page will appear.
4. Press Select next to **Bank** and select the bank from the list.

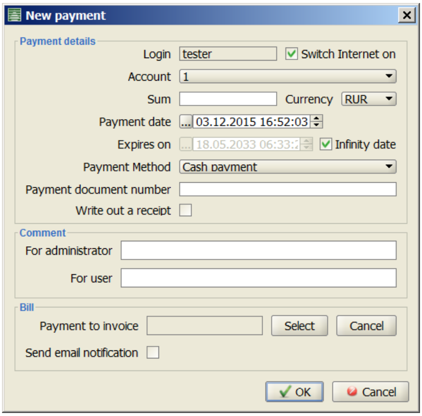


5. Press Ok to close the list of banks.
6. Press Ok in the user properties window to save the changes.

Making payment

To make a payment to a particular account of the given user:

1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.



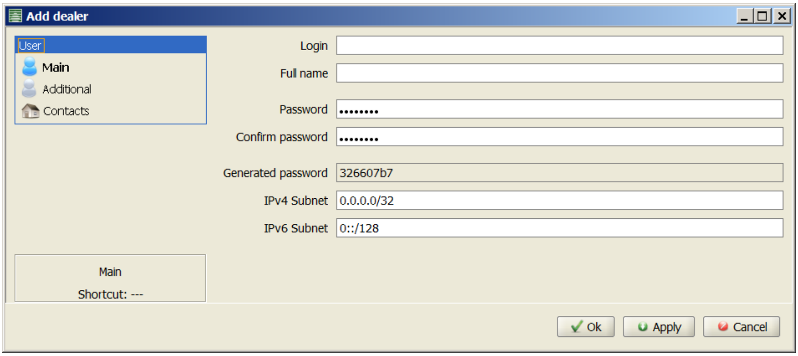
Login: tester
☑ Switch Internet on
Account: 1
Sum:
Currency: RUR
Payment date: 03.12.2015 16:52:03
Expires on: 18.05.2033 06:33:
☑ Infinity date
Payment Method: Cash payment
Payment document number:
Write out a receipt ☐
Send email notification ☐

2. Select the user in the list and press **New payment**. The payment details window will show up.
3. If necessary, select another account from the drop-down list.
4. Select the payment currency from the list.
5. Enter the sum of payment.
6. Enter the payment date or leave the default value (current date).
7. Enter the payment expiration date or leave the default value (never).
8. Enter the arbitrary comments for the administrator and for the user.
9. Select the payment method from the list.
10. If the payment is being done on demand of some external document, enter the number of that document.
11. If the payment is being done on demand of some internal invoice, select the number of that invoice from the list.
12. Press **✓ Ok** to finalize the payment.

Creating dealers

To create a new dealer:

1. Press **Dealers** on the left pane under **Users & Groups**. A list of registered dealers will appear.
2. Press **Add**. The dealer properties window will show up.



Login:
Full name:
Password: ••••••••
Confirm password: ••••••••
Generated password: 326607b7
IPv4 Subnet: 0.0.0.0/32
IPv6 Subnet: 0::/128

3. Enter the dealer’s login and the full name.

4. Copy the generated password for handing it over to the operator, or enter the new password twice.
5. If appropriate, specify a subnet to allow the dealer's authorization from.
6. Press ✓ Ok to finalize the creation of the new dealer.

Setting dealer's permissions

To permit dealer's access to certain system objects:

1. Press **Dealers** on the left pane under **Users & Groups**. A list of existing dealers will appear.
2. Select the dealer in the list and press Edit. The dealer properties window will show up.
3. On the left pane of the dealer properties window open **Users** under **Permissions**. The list of users will appear.
4. In the **Permissions granted** column mark the check boxes corresponding to the users of your choice, in order to enable the dealer with the access to these users.

Note that the users (unlike other entities) are attached to dealers in exclusion mode, i.e. each user may be attached to only one dealer. The users already attached to other dealers have their permission check boxes disabled.

5. In a similar way set up the dealer's access to other entities on the pages **Accounting periods**, **Services**, **Tariffs**, and **Houses**.
6. Press ✓ Ok to save the changes.

Linking users to dealers

Besides the way described in **Setting dealer's permissions** on page **141**, the dealer's access to a user may be set up as follows:

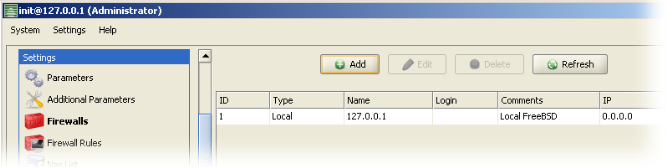
1. Press **Users** on the left pane under **Users & Groups**. A list of existing users will appear.
2. Select the user in the list and press Edit. The user properties window will show up having by default its **Main** page open.
3. Press **Link to dealer**. A list of registered dealers will show up.
4. Select the dealer from the list and press ✓ Ok to link the user to the dealer.
5. Close the user properties window.

The **Link to dealer** button with similar functionality may also be found on the search page (see **Search page** on page **107**).

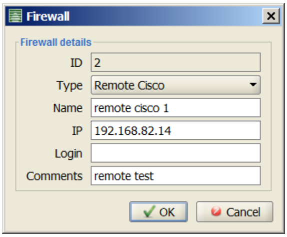
Creating firewalls

To create a new firewall:

1. Press **Firewalls** on the left pane under **Settings**. The list of existing firewalls will appear.



ID	Type	Name	Login	Comments	IP
1	Local	127.0.0.1		Local FreeBSD	0.0.0.0



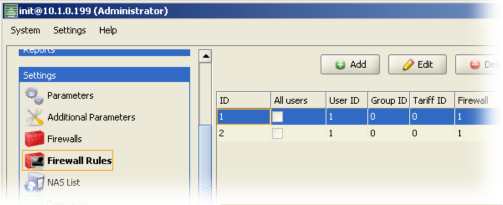
ID 2
Type Remote Cisco
Name remote cisco 1
IP 192.168.82.14
Login
Comments remote test

2. Press Add. The **Firewall** window will show up.
3. In the **Type** drop-down list select **Local** if the commands are going to be executed locally, or **Remote Cisco** for execution over rsh.
4. Enter the firewall **Name**. Make sure that no other firewall with the same name exists in the system.
5. If the firewall is about to be used in the properties of IP traffic service link as **NetFlow provider**, enter its IP address in the **IP** input field.
6. If **Type** was set to **Remote Cisco**, enter **Login** for rsh authorization.
7. Enter arbitrary **Comments**.
8. Press ✓ Ok. New firewall will be created.

Creating firewall rules

To create a new firewall rule:

1. Press **Firewall rules** on the left pane under **Settings**. The list of registered firewall rules will appear.



ID	All users	User ID	Group ID	Tariff ID	Firewall
1	☐	1	0	0	1
2	☐	1	0	0	1

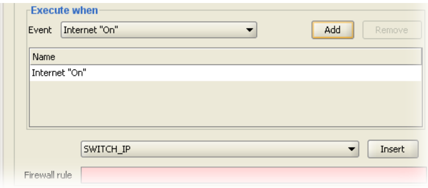
2. Press Add. The **Firewall rules** window will show up.
3. Enter **Comment** to be able to tell this rule from the others in the list.

4. In the **Firewall** drop-down list select the firewall to run the command on, or **Any** to apply the command to all firewalls available at the time of its execution.
5. In the **Execute for** group either set the **All users** check box, or set one or more of the following conditions to define the applicability domain:
   - **User ID**;
   - **Group name** (select from the drop-down list);
   - **Tariff name** (select from the drop-down list).

   If more than one condition is checked, their union is used. To use the intersection of conditions, check **All parameters match**.

*Firewall rules associated with the global system events (**Raw traffic file closed** and **Log file closed**) require that the **All users** option must be checked.*

1. In the **Execute when** group select one or more events to initiate the rule in question using the drop-down list of events and the **Add** and **Remove** buttons.
2. Enter the command template in the **Firewall rule** field. Use the necessary variables by selecting them from the drop-down list (see **UTM5 RFW: Variables** on page **187**) and pressing **Insert**. On execution the variables are substituted with their corresponding values. The set of available variables depends on the selected initiating events (see the list in **UTM5 RFW: Events** on page **190**). Attempts to use the non-available variables cause warnings.



Execute when

Event: Internet "On"

Name
Internet "On"

SWITCH_IP

Firewall rule:

3. Press ✓ Ok. New firewall rule will be created.

9

UTM5 CORE

Introduction

Core of the billing system is a basic module responsible for the database access. The core provides access to it and processes incoming information under the internal rules (such as tariffication, periodical charge-offs, etc). The core is a separate multithreaded process functioning in a user-operated mode. At the startup the core demands administrator's privileges. Due to its structure the core blends in with multiple-processor architectures and uses all resources evenly at high workloads.

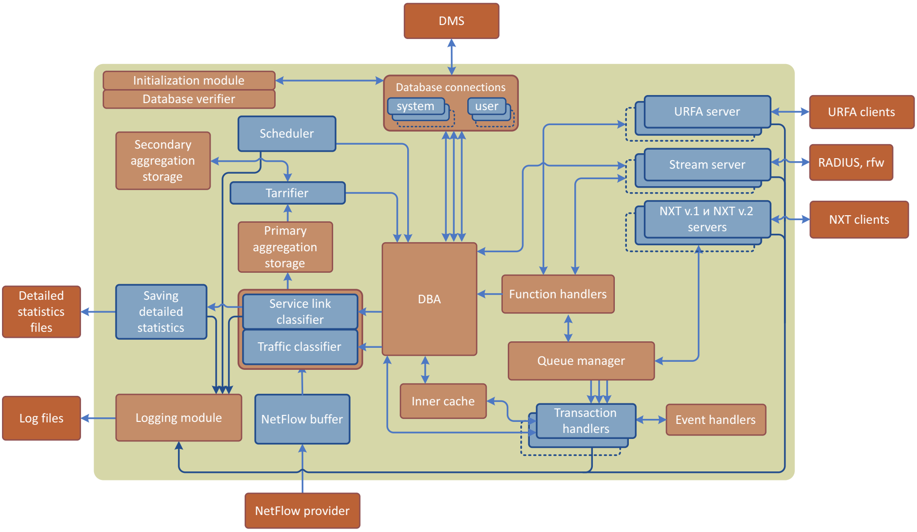


Рис. 1. UTM core schematics (since v. 5.3-001)

Main Components of the Core

URFA request handler (UTM Remote Function Access) is a server that invokes remote procedures. It receives connections of clients and executes requested commands in the core. This component serves mainly as an organizer of the user and administrator interfaces.

NetFlow buffer receives traffic data in NetFlow format version 5, 7, and 9. Devices that do not support statistics delivery via these protocols must rely on some auxiliary utility to convert their output into compatible format.

Traffic classifier is a core module that sorts traffic into classes according to characteristics defined in system settings. These characteristics may be specified via the UTM control center.

The business logic module is responsible for tariffication of all services, including IP traffic transmission. It converts amount of services consumed into monetary equivalent, taking into account all dependencies defined by system administrator.

The log file keeps all records of UTM functions. It provides the administrators with all sorts of diagnostic information about system failures.

The database access module is a united DB interface which transfers intrasystem data requests into requests to an external database. It is an abstraction level giving the UTM independence of any particular DBMS.

Data are received via NFBuffer and URFA. Input data are read from the database when the system is launched. Changes made directly in the database afterwards may cause uncontrolled behavior of the system.

NetFlow data go to the business module where they are processed and all necessary charge-offs are calculated. At peak workloads NetFlow can be buffered to reduce possible losses. Raw NetFlow data are stored in files. At startup this DB module is started in a separate thread with (if possible) high priority.

Startup

UTM5 core executable file is `/netup/utm5/bin/utm5_core`.

Possible command line parameters are:

`-p <path>`	Path to the PID file
`-c <path>`	Path to the config file
`-v`	Version number and parameters information

The following options for `utm5_core` startup are available:

1. Direct start of the `utm5_core` executable with necessary parameters;
2. Start on watchdog with `start` parameter:

```
/netup/utm5/bin/safe_utm5_core start
```

The script will restart `utm5_core` automatically on failure;

1. Start via the automatic startup script (recommended).
   On Linux:

```
/etc/init.d/utm5_core start
```

On FreeBSD or Solaris:

```
/usr/local/etc/rc.d/utm5_core.sh start
```

To stop the `utm5_core` and the watchdog script, execute:

on Linux –

```
/etc/init.d/utm5_core stop
```

on FreeBSD or Solaris –

```
/usr/local/etc/rc.d/utm5_core.sh stop
```

Core settings

System core parameters may be set up in the following ways:

- Via the config file;
- Via the administrator's interface (see **Interface parameters** on page **152** for more detail).

Config file parameters are used during the initialization of the system core and other components. Any changes to these parameters are applied after the next restart. Interface parameters, on the contrary, are related to the system's behavior after startup and may be changed at any moment, unless stated otherwise. The changes are applied immediately.

Config file

Config file used by the UTM5 system core on Unix platforms is located at `/netup/utm5/utm5.cfg`. For Win32 version, it is `utm5.cfg` at the installation directory (which by default is `C:\Program Files\NetUP\UTM5\`).

Config file has the following format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered a comment.

Below is the list of all possible parameters.

Database-related parameters:

Parameter	Possible values	Default value	Description
`database_type`	mysql, postgres	Mandatory parameter	Database type
`database`	String	Mandatory parameter	Database name
`database_host`	String	`localhost`	Database host address
`database_login`	String	current user’s login	Database access login
`database_password`	String	empty string	Database access password
`database_sock_path` (a)	String	`/tmp/mysql.sock`	Path to a unix-socket used for the database server connection. Should be used only for MySQL database and only if `database_host` is not defined or is equal to `localhost`.
`database_port` (a)	String	3306	Port number for database access
`dbcount`	Number from 2 to 64	6	Number of database connections open simultaneously by the billing system core for user operations
`dbcount_sys`	Number from 2 to 64	4	Number of database connections open simultaneously by the billing system core for system operations
`database_reconnect_count`	Integer number	5	Number of database connection attempts in case of failure. Also, the number of repeated SQL requests in case of failure

Parameter	Possible values	Default value	Description
`database_reconnect_sleep`	Integer number	2	Delay in seconds before repeated connection attempt or SQL query
`database_charset` [a]	Encoding specification string	`utf8`	Database connection encoding
`verify_database`	enable, disable	enable	Verify database before starting the UTM5 core
`verify_archive_tables`	enable, disable	disable	If the database verification is enabled, also verify archived tables
`verify_database_index`	enable, disable	disable	Verify indexes before starting the UTM5 core

[a] Relevant for MySQL only.

URFA-related parameters:

Parameter	Possible values	Default value	Description
`urfa_bind_host` [a]	IP address of the interface, or 0.0.0.0	0.0.0.0 (server off)	IP address of the server listening to URFA requests
`urfa_bind_port`	Number from 1 to 65534	11758	Port listening to URFA requests

[a] Multiple instances of the parameter are possible.

Stream-related parameters:

Parameter	Possible values	Default value	Description
`stream_bind_host`	IP address of the interface, or 0.0.0.0	0.0.0.0	IP address of the server listening to Stream requests
`stream_bind_port`	Number from 1 to 65534	12758	Port listening to Stream requests

NXT-related parameters:

Parameter	Possible values	Default value	Description
`nxt_bind_host`	IP address of the interface, or 0.0.0.0	0.0.0.0	IP address of the server listening to NXT v.1 requests
`nxt_bind_port`	Number from 1 to 65534	11777	Port listening to NXT v.1 requests
`nxt_v2_bind_host`	IP address of the interface, or 0.0.0.0	0.0.0.0	IP address of the server listening to NXT v.2 requests
`nxt_v2_bind_port`	Number from 1 to 65534	11778	Port listening to NXT v.2 requests
`iptv_cluster_host`	IP address	Not set	NetUP cluster core IP address
`iptv_cluster_port`	Number from 1 to 65534	50500	Port that IPTV cluster core is listening to for incoming connections

NetFlow buffer parameters:

Parameter	Possible values	Default value	Description
`nfbuffer_host`	String	0.0.0.0	IP address of the server listening to NetFlow stream
`nfbuffer_port`	String	9997	Port listening to NetFlow stream
`nbuffer_bufsize`	Integer number	Set by OS	Size of the UDP socket buffer used to accept the N-etFlow stream

Traffic counting parameters:

Parameter	Possible values	Default value	Description
`-classifier_-traffic_file`	Path to file	`/netup/utm5/db/ traffic.dat`	File to store traffic information when UTM5 core stops

Document generation parameters:

Parameter	Possible values	Default value	Description
`doc_path`	File path	/netup/utm5/doc	*.odt file storage directory
`tmp_path`	File path	/tmp	Temporary files storage
`libreoffice_path`	File path	/usr/bin/libreoffice	LibreOffice executable path
`max_upload_size`	Size in bytes	1000000	Maximum size of document template / contract file for upload

Logging parameters (for more details see **System description: Logging** on page **22**):

Parameter	Possible values	Default value	Description
`log_level`	number from 0 to 3	1	Level of messages to be written to the log file
`log_file_main`	Path to file	standard error stream	Main log file
`log_file_debug`	Path to file	standard error stream	Debugging log file
`log_file_critical`	Path to file	standard error stream	Critical log file
`log_file_v-erificator`	Path to file	`/netup/utm5/log/ verificator.sql`	Database verifier log file
`core_pid_file`	Path to file	`/var/run/utm5_core.pid`	PID file
`rotate_logs`	yes, on, enable	not set (rotation off)	Enables log files rotation
`max_logfile_count` [a]	Number	Not set (unlimited)	Maximum number of log files to keep
`max_logfile_size` [a]	Size in bytes	10485760	Maximum size of log file
syslog_name	string	Not set	Log entry prefix (when logging to syslog is enabled)

[a] Works if log file rotation is enabled.

Stack parameters:

Parameter	Possible values	Default value	Description
`thread_stack_size`	Size in bytes (not less than 65536)	8388608	Business logic thread stack size
`rpc_stack_size`	Size in bytes (not less than 65536)	Not set	URFA server thread stack size

License parameters:

Parameter	Possible values	Default value	Description
`ssl_cert_file`	Path to file	`/netup/utm5/cert.crt`	Certificate file
`ssl_privkey_file`	Path to file	`/netup/utm5/privkey.pem`	Private key file
`ssl_privkey_passphrase`	String	empty string	Private key password

Interface parameters

Interface parameters set up via the administrator interface are displayed and set up in the **Settings: Parameters** interface window. The **Variable** field contains parameter’s name, and **Value** contains its value. It is possible to edit existing parameters as well as set up new ones.

Below is the list of available parameters affecting the UTM5 core.

Detailed statistics parameters:

Parameter	Possible values	Default value	Description
`raw_records_send_limit`	Integer number	100000	Maximum number of records returned by the detailed traffic report function
`raw_buffer_size`	Size in bytes	1000000	Size of the internal buffer for detailed statistics
`raw_commit_interval`	Integer number	5	Periodicity (sec.) of dumping the buffer to disk
`raw_max_files`	Integer number	10	Maximum number of files with primary information on traffic. If exceeded, old files are removed automatically
`raw_max_size`	Size in bytes > 5242880	100000000	Maximum size of file with primary information on traffic. If exceeded, the file closes automatically and the output switches to the new file
`raw_prefix`	Folder name	`/netup/utm5/db`	Name of folder to store the files with primary information on traffic
`raw_fd_process_script`	Path to executable file	`/netup/utm5/bin/raw_fd_script` [a]	Script to execute on closing of the file with primary traffic information. Name of the file being closed is passed to the script as a parameter

a. In UTM5 assembled for Win32 the parameter is not set by default.

Traffic counting parameters:

Parameter	Possible values	Default value	Description
`bytes_in_kbyte`	Integer number	1024	Number of bytes in kilobyte. Square of this number stands for number of bytes in megabyte
`traffic_mult_coef`	Real number	1	Correction factor for traffic quantity

Traffic aggregation parameters:

Both parameters act simultaneously, i.e. a charge-off is performed whenever any of the two conditions is reached: passing of the specified time or accumulation of the specified total cost.

Parameter	**Possible values**	**Default value**	**Description**
`traffic_aggregation_interval`	Time in seconds	900	Time before charge-off for the aggregated traffic
`aggregation_todisc_barrier`	Positive real number	5	Total cost to achieve before charge-off for the aggregated traffic

Smaller values of aggregation parameters would cause faster growth of charge-off tables, which constitute a bulky part of the database. This, in turn, may require special measures, see ***Archiving of tables*** *on page* ***270****.*

Parameters related to smooth charge-off of the periodic costs:

Parameter	**Possible values**	**Default value**	**Description**
`discount_barrier`	Real number > 0.0001	0.01	Threshold for blocking charge-off of smaller sums so as to prevent accumulation of rounding errors
`flow_discount_per_period`	Integer number > 5	64	Minimal number of charge-offs for periodic services per period

Default parameters for newly created accounts:

Parameter	**Possible values**	**Default value**	**Description**
`default_vat_rate`	Real number	0	Default VAT rate value for the newly created account

Parameters related to automatic user registration:

Parameter	**Possible values**	**Default value**	**Description**

`card_callback_enable`	0, 1, 2, or 3	0	Sets default value of the **Callback enabled** and **Ringdown enabled** parameters for dial-up service with automatic registration:
			0 – **Callback enabled** is set;
			1 – both are set;
			2 – both are unset;
			3 – **Ringdown enabled** is set

Parameter	Possible values	Default value	Description
`default_dialup_cid`	String	Not set	Sets default value of the Allowed CIDs" parameter for dial-up service with automatic registration
`default_dialup_csid`	String	Not set	Sets default value of the "Allowed CSIDs" parameter for dial-up service with automatic registration
`card_tel_uid_len`	Integer number	Whole PIN	Sets the length of PIN portion used as login for dial-up service with automatic registration; the rest is used as password
`card_user_prefix`	String	`card_`	Sets the prefix for card users' logins (see **Prepaid cards** on page **28**)

Template-related parameters:

Parameter	Possible values	Default value	Description
`templates_prefix`	Path	`/netup/utm5/templates`	Path where the template files are stored

Miscellaneous parameters:

Parameter	Possible values	Default value	Description
`hotspot_refresh_timeout`	Integer number	300	Time of validity since last refresh for hotspot sessions, in seconds
`system_currency`	Integer number	810	System currency ID. If set to 810, online refresh of exchange rates is enabled
`web_session_timeout`	Integer number	300	Maximum time to keep the unique key (SID) of URFA session
`null_prepaid_traffic_if_tariff_change`	1	Not set	If set to 1, prepaid traffic is zeroed when switching a tariff plan
`lite_search_ent`	Integer number	5	Maximum number of records returned by the search when called from the interface
`login_prefix_separator`	Character	:	Delimiter between the login prefix and the card number or PIN for autoregistered users. Not recommended to change
`tel_report_dont_show_id`	Arbitrary	Not set	Switches off the zone indicator in telephony reports at the user's web interface

Parameter	Possible values	Default value	Description
`legacy_invoices`	1	Not set	Switches off the aggregation of different types of traffic in invoices
`enchanced_telephony_invoice_algorithm`	1 or 0	0	If set to 0, all telephony charges are gathered in one invoice; any other va-lue implies separate invoices from different telephony operators
`encypt_passwords`	yes, on, enable	Not set	Enables user, system user password and passwords used in service links encryption

Mail-related parameters:

Parameter	Possible values	Default value	Description
`smtp_relay`	IP address or a host name	not set	IP address of the SMTP server to relay mail messages with
`smtp_port`	Number from 1 to 65534	25	SMTP server port
`smtp_fqdn`	String	localhost	Destination domain name for e-mail
`smtp_sender`	E-mail	utm5_core	Default sender address for e-mail
`smtp_recipient`	E-mail	root	Default recipient address for e-mail
`invoice_-subject`	String	Invoice	Default subject for invoice e-mails
`invoice_text`	String	Invoice message	Default text for invoice e-mail messages
`notification_borders`	Space-separated list of real numbers	0	Threshold values of account balance to notify the user by e-mail when crossed
`notification_message`	String with variables (1)	empty string	Default text for balance notification e-mail messages
`notification_message_-subject`	String	empty string	Default subject for balance notification e-mail messages

Parameter	Possible values	Default value	Description
`notification_message_from`	E-mail address	utm5_core	Default `From:` address for balance notification e-mail messages
`balance_notification_e-mail`	E-mail address	root	Default recipient address for balance notification e-mail messages
`payment_notification_message`	String with variables (2)	Your payment succeeded! Payment_Sum = AMOUNT Payment id = PAYMENT_ID	Default text for payment notification e-mail messages
`payment_notification_subject`	String	Payment	Default subject for payment notification e-mail messages

1. Variables to use in `notification_message`:

Variable	Description
`FULL_NAME`	Client name
`ACCOUNT_ID`	Client's main account ID
`BALANCE`	Client's main account balance at the time of message preparation
`DATE`	Date of message preparation
`EMAIL`	E-mail address for the notification to be sent to

2. Variables to use in `payment_notification_message`:

Variable	Description
`FULL_NAME`	Client name
`ACCOUNT_ID`	Client's main account ID
`AMOUNT`	Payment sum in internal currency
`PAYMENT_ID`	Payment ID

Parameters related to system messages:

Parameter	Possible values	Default value	Description
`web_message_group`	Number	Not set	ID of the system group to whose members (or to all system users, if not set) the user's system messages are relayed

Parameter	Possible values	Default value	Description
`notification_message_by_wintray`	“Yes”	Not set	Enables balance notification via system messages sent to users when their balance passes over the notification borders

UTM5 RADIUS parameters:

Parameter	Possible values	Default value	Description
`radius_max_session_age`	Time in seconds	86400	Maximum age of open sessions to be loaded to the R-ADIUS server on startup. If set to 0, none are loaded
`radius_do_accounting`	1	Not set	Enables tariffication by Stop packets
`radius_do_interim_accounting`	1	Not set	Enables tariffication by Interim-update packets
`radius_realm` [a]	String	Not set	Suffix to be chopped off the login

a. Multiple instances of the parameter are possible.

UTM5 RFW parameters:

Parameter	Possible values	Default value	Description
`fw_rule_offset`	Integer number	5000	Number to be added to the user ID to obtain the `RULE_ID` value

Parameters related to external payment systems (see **Payment systems** on page **277**):

Parameter	Possible values	Default value	Description
`ext_payment_inet_on`	String	Not set	If set to non-empty value, turns on the **Switch internet on** option (see **Payment page: Switch internet on** on page **106**) for the payments coming from external payment systems via the integration module
`ext_payment_notify`	String	Not set	If set to non-empty value, turns on the **Send notification** option (see **Payment page: Send email notification** on page **107**) for the payments coming from external payment systems via the integration module

Other parameters are deprecated and highly non-recommended, unless stated otherwise.

10

UTM5 RADIUS

Introduction

NetUP RADIUS server is an application intended for real-time processing of the incoming requests using Remote Authentication Dial In User Service (RADIUS) protocol as described in RFC 2865, RFC 2866 and RFC 5176.

RADIUS protocol serves for secure authorization, authentication and accounting between NAS and authorization servers. Besides that, the protocol is applicable to collect information on consumed services, such as connection time, amount of traffic, user’s IP address, etc.

UTM5 RADIUS interacts with the UTM5 core using Stream protocol.

One instance of UTM5 core may work with only one RADIUS server.

RADIUS server is a part of several different modules that require separate licenses. To verify the availability of the licenses and their terms of validity, see **About: Licenses** in the UTM5 administrator’s interface and check for at least one of the following items in the list: **VPN/Dial-up module**, **Telephony module**, or **Hotspot module**.

RADIUS protocol description

RADIUS protocol has been developed for easy administration of huge modem pools. For example, when the network contains several devices accessed by users, with each device storing data of all users, it becomes extremely tricky to administrate such a system. The problem may be solved by introducing one central authorization server and make all network devices send requests to it using the standard RADIUS protocol. In this case, device of any RADIUS-supporting manufacturer may function as NAS.

In case if the NAS is intended to interact with UTM5 RADIUS over the RADIUS protocol, it does not keep its user base. On user’s connection, NAS makes an Access-Request call. UTM5 RADIUS considers whether to permit a connection, and responds to the NAS with Access-Accept on positive decision or with Access-Reject otherwise.

If the decision require additional information exchange, Access-Challenge is sent to the NAS.

```
 -+-+-+-+-+-+-+-                            -+-+-+-+-+-+-+-+-+-
 |            |     ---->Access-Request     |                 |
|    NAS     |     <----Access-Challenge   |  UTM5 RADIUS     |
|            |     ---->Access-Request     |                  |
|            |     <----Access-Accept      |                  |
+-+-+-+-+-+-+-+                            +-+-+-+-+-+-+-+-+-+
```

In case if NAS is configured to send the connection info, after establishing a connection UTM5 RADIUS sends an Accounting-Request. Depending on the configuration, NAS may also send additional periodical Accounting-Requests containing current status info on the connection. When a connection is broken, NAS must send a summarizing Accounting-Request, given that some Accounting-Requests for this connection have already been exchanged before.

On receiving an Accounting-Request, UTM5 RADIUS creates, changes, or removes an object associated with the given connection. Depending on the Accounting-Request contents, some additional actions may be performed to maintain the delivered information on the connection.

After successfully handled an Accounting-Request, UTM5 RADIUS sends to NAS a confirmation Accounting-Response. On failure, no response is sent.

Packets from an unknown (not registered) NAS are ignored.

The interaction between UTM5 RADIUS and NAS is performed in RADIUS packets sent by UDP protocol. Commonly, port 1812 is used by UTM5 RADIUS to receive Access-Requests, and port 1813 to receive Accounting-Requests.

Generally a RADIUS packet contains the following fields:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Code           | Identifier       | Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                       |
|                 Authenticator                         |
|                                                       |
|                                                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attributes...
+-+-+-+-+-+-+-+...
```

- **Code –** is an one-byte field used to identify the RADIUS packet type. UTM5 RADIUS supports the following types:

Type	Name	Sent or received by UTM5 RADIUS
1	Access-Request	Received
2	Access-Accept	Sent
3	Access-Reject	Sent
4	Accounting-Request	Received
5	Accounting-Response	Sent
11	Access-Challenge	Sent

- **Identifier –** is a one-byte field intended to relate the request to the response. Duplicate requests with the same ID coming from the same NAS shortly after each other are ignored.
- **Length –** is a two-bytes field containing packet size.
- **Authenticator –** is a 16-bytes field that contains data for checking the packet's authenticity. For a request, it is some unique sequence used together with the md5 of the secret word common for the UTM5 RADIUS and NAS for reversible encoding of the user’s password. For a response, it is md5 of **Code**, **Identifier**, **Length**, **Authenticator**, and **Attributes** fields together with the secret word.

The common secret word must be considerably hard to break. It is strongly not recommended to leave it blank. UTM5 RADIUS uses the sender address of the RADIUS packet to derive the common secret word.

- **Attributes –** is a variable-length field containing the list of RADIUS attributes.

Each RADIUS attribute contains specific information on a request or a response. Generally, a RADIUS attribute looks as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Type            | Length          | Value...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...
```

- **Type –** is a number describing the attribute type. Possible types are listed in RFC 1700.
- **Length –** is the summary length of the **Type**, **Length**, and **Value** fields.
- **Value –** is the type-specific information. Depending on the type, may contain the following fields:
  - **text –** from 1 to 253 bytes of UTF-8 text, zero byte forbidden;
  - **string –** from 1 to 253 bytes of binary info;
  - **address –** is a 32-bit data interpreted as an address;
  - **integer –** is a 32-bit data interpreted as unsigned integer;
  - **time –** is a 32-bit data interpreted as time in seconds since 00:00:00, January 1, 1970 UTC.

Some attributes may appear in a packet more than once, which is interpreted in a type-specific manner. Order of attributes is important.

From this point on, the RADIUS attributes are referred by common name followed by the type ID in parentheses, for example: User-Name (1).

There is a certain attribute type Vendor-specific (26) designed to store extended vendor-specific data. These data are interpreted as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Vendor-Id                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Vendor-Type    | Vendor-Length    | Data...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...
```

- **Vendor-Id –** is a number describing the organization that defines this attribute (for more details see RFC 1700).
- **Vendor-Type –** is a number that described the attribute meaning.
- **Vendor-Length –** is the summary length of the **Vendor-Type**, **Vendor-Length** and **Data** fields.
- **Data –** contains the actual data.

From this point on, the vendor-specific attributes are referred by common name followed by the semicolon-separated **Vendor-Id** and **Vendor-Type** in parentheses, for example: Cisco-AVPair (9;1).

DM and CoA requests

When session parameters need to be changed or a session needs to be terminated, UTM5 RADIUS may send CoA (Change-of-Authorization) or DM (disconnect-message) requests according to RFC 5176. UTM5 sends a separate request for each session for changing parameters or terminating a session.

DM and CoA requests have the same format and are send to the 3799 UDP port. This port may be changed in NAS settings (see **Administrator’s interface: NAS list** on page **87**).

DM requests are used to terminate session if user’s account balance becomes negative. The following attributes might be used to identify a session:

- **User-Name –** is a user name, associated with one or more sessions
- **NAS-Port –** is a port used by session that needs to be terminated
- **Framed-IP-Address –** is an IPv4 address associated with the session
- **Vendor-Specific –** is one or more vendor-specific attributes
- **Called-Station-Id –** is the called party identifier

- **Calling-Station-Id –** is the calling party identifier
- **Acct-Session-Id –** is an ID that let’s one clearly identify a session on a NAS

NAS replies with a Disconnect-ACK in case it was able to identify and terminate the session, otherwise it replies with a Disconnect-NAK.

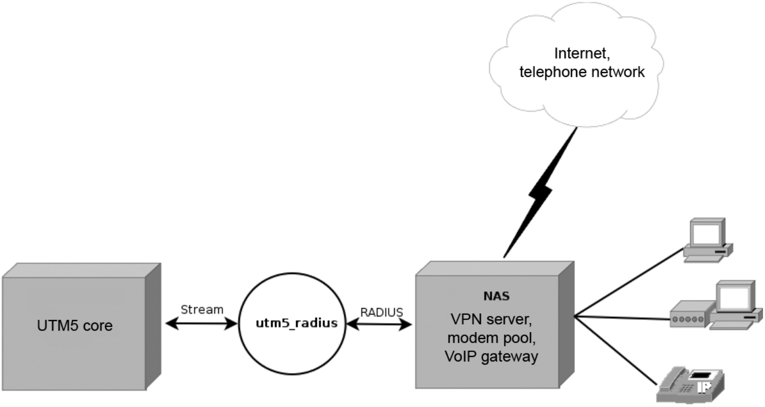
CoA requests are used for changing shaping parameters for current session (for more information see **UTM5 Dynashape: Workflow description** on page **199**). These parameters are most likely to be changed at a certain time or after reaching a certain traffic limit. In case NAS supports CoA requests and a corresponding flag is checked in the administrator’s interface, if one of these happens, UTM5 RADIUS will send a CoA request to NAS.

Like a DM request, CoA request contains attributes required for identifying a session and the new values of RADIUS parameters that need to be updated (see **UTM5 Dynashape: RADIUS parameters** on page **200**). If NAS is able to identify session and update RADIUS parameters it replies with a CoA-ACK, otherwise it replies with a CoA-NAK.

Workflow description

UTM5 RADIUS works as follows:

1. Connects to the UTM5 core.
2. Retrieves from UTM5 the info on events to await.
3. Interacts with NAS.
4. Sends the resulting data to UTM5.



On startup UTM5 RADIUS connects to the UTM5 core, authorizes according to its config file parameters, and establishes a Stream connection. Once a connection is set, the UTM5 core passes to UTM5 RADIUS the description of the needed objects in the corresponding events.

UTM5 RADIUS keeps the Stream connection to the core. Upon creating, changing, or deleting system objects related to UTM5 RADIUS functionality, the UTM5 core sends the corresponding event over Stream in order to inform UTM5 RADIUS.

On reception of certain events, UTM5 RADIUS creates, modifies or deletes its inner records related to the following objects:

- IP groups;
- NAS;
- Accounts;
- Time ranges;
- IP traffic, hotspot, and telephony services;
- IP traffic, hotspot, and telephony service links;
- Telephone zones;
- Telephone directions;
- IP pools.

Authorization

On receiving an Access-Request UTM5 RADIUS performs the following:

1. User authentication with one of the methods:
    - PAP
    - CHAP
    - MS-CHAP v1
    - MS-CHAP v2
    - EAP-MD5
    - EAP-TTLS
    - Digest

The Digest authentication is implemented according to **http://tools.ietf.org/id/draft-sterman-aaa-sip-00.txt**, *rather than the corresponding RFC.*

The authorization request must contain the User-Name (1) attribute. The portion of its value before ':' is interpreted as the Callback_prefix parameter, and the rest proceeds to the next step. All letters contained in the login are cast to lower case. If the User-Name (1) attribute is missing, the Access-Request is ignored and the rest of actions skipped.

If authentication fails or requires unsupported method, an Access-Reject is sent to NAS.

If the `guest_pool_name` parameter is set in the UTM5 RADIUS config file (see **Config file** on page **170**), the guest users may be authorized as well.

1. Using the login from step **1**, a corresponding service link is found. The following actions depend on the service link type.

- For an IP traffic service link:
  - If the `radius_auth_vap` parameter is set in the UTM5 RADIUS config file, the account referred in the given service link is checked for blocking.
  - The given IP group is checked for presence of free IP addresses.
  - If the IP group parameter **Allowed CID** is not empty, the Calling-Station-Id (31) value is checked against it as a regular expression.
  - If the `radius_nas_port_vpn` parameter is set in the UTM5 RADIUS config file, the NAS-Port-Type (61) value is checked to match one of its values.

  Failure of any of these checks result in sending an Access-Reject packet and skipping the rest of actions.

  If the checks are successful, an Access-Accept packet is sent with the following attributes:
  - Service-Type (6) set to 2.
  - Framed-IP-Netmask (9) set to `0xFFFF FFFF`.
  - Framed-Routing (10) set to 0.
  - Framed-Protocol (7) set to1.
  - Framed-IP-Address (8) set to the first free IP address in the given IP group. The IP address is marked busy for the time span defined by the parameter `radius_ippool_acct_timeout` from the UTM5 RADIUS config file.
  - Session-Timeout (27) set to the value of `radius_default_session_timeout` parameter from the UTM5 RADIUS config file.
- For a hotspot service link:
  - The presence of the Framed-IP-Address (8) in allowed networks set in the service properties (skip if the list is empty) is checked.
  - The amount of connections established for this service link is compared to the maximum number of simultaneous connections set in the service properties.
  - The given account is checked for blocking.
  - Maximum connection time is calculated from the account balance and the service link parameters.

  Failure of any of these checks result in sending an Access-Reject packet and skipping the rest of actions.

  If the checks are successful, an Access-Accept packet is sent with the following attributes:
  - Mikrotik-Xmit-Limit (14988;2) set to the maximum session time.
  - Mikrotik-Rate-Limit (14988;8) set to the BandwithLimit parameter.
- For a dial-up service link:
  - The Callback_prefix parameter is checked for consistency with the **Callback allowed** and **Ringdown allowed** parameters. That is, if **Callback allowed** is not set, Callback_prefix must not be set; on the contrary, if **Ringdown allowed** is not set, Callback_prefix must be set.

- If **Allowed CID** is not empty, the Calling-Station-Id (31) value is checked against it as a regular expression.
- If **Allowed CSID** is not empty, the Called-Station-Id (30) value is checked against it as a regular expression.
- The amount of connections established for this service link is compared to the maximum number of simultaneous connections set in the service properties.
- If the `blocked_pool_name` parameter is not set in the UTM5 RADIUS config file, the given account is checked for blocking.
- Maximum connection time is calculated from the account balance and the service link parameters.
- If a registered IP pool is set in the service properties, it is checked for presence of free addresses.
- If the `radius_nas_port_vpn` parameter is set in the UTM5 RADIUS config file, the NAS-Port-Type (61) value is checked to match one of its values.

Failure of any of these checks result in sending an Access-Reject packet and skipping the rest of actions.

If the checks are successful, an Access-Accept packet is sent.

If the user is blocked and the `blocked_pool_name` parameter is set in the UTM5 RADIUS config file, the IP address will be issued from the pool intended for blocked users, which is defined by this parameter.

When user's account is unblocked, a DM (disconnect message) will be sent in order to break the session. After that, the customer will require to reconnect

If the user is not registered and the `guest_pool_name` parameter is set in the UTM5 RADIUS config file, the IP address will be issued from the pool intended for guest users, which is defined by the `guest_pool_name` parameter.

If the dial-up service parameter **Pool name** is set to some registered IP pool, the following attributes are returned:

- Service-Type (6) set to 2.
- Framed-IP-Netmask (9) set to `0xFFFF FFFF`.
- Framed-Routing (10) set to 0.
- Framed-Protocol (7) set to 1.
- Framed-IP-Address (8) set to the free IP address from the given IP group. The IP address is marked busy for the time span defined by the parameter `radius_ippool_acct_timeout` from the UTM5 RADIUS config file.
- Session-Timeout (27) set to the maximum session time.

Otherwise, i.e. if the dialup service parameter **Pool name** is set to some IP pool not registered in the system, the following attributes are returned:

- Service-Type (6) set to 2.
- Framed-MTU (12) set to 1500.

- Framed-Routing (10) set to 0.
- Framed-Protocol (7) set to 1.
- Session-Timeout (27) set to the maximum session time.
- Cisco-AVPair (9;1) set to `addr-pool=<pool name>`.

Besides that, if Callback_prefix is set, the following attributes are added:

- Callback-Number (19) set to callback number, if the UTM5 RADIUS config file parameter `radius_callback_avpair_enable` is not set.
- Callback-Id (20) set to Callback login, if the UTM5 RADIUS config file parameter `radius_callback_avpair_enable` is not set.
- Cisco-AVPair (9;1) set to `lcp:callback-dialstring=<callback_ - prefix>`, if `radius_callback_avpair_enable` is set.

The issued IP address is marked busy for the time span defined by the parameter `radius_ippool_timeout` from the UTM5 RADIUS config file.

For more details on telephony module, see ***IP telephony module*** *on page* ***257****.*

For any type of service, the NAS attributes set for the service link are included in the Access-Accept response.

Accounting

Accounting-Requests are used by UTM5 RADIUS to determine if an IP address is occupied, charge for hotspot, dial-up or telephony services, charge for consumed traffic and dynamically create, modify or remove IP groups.

Accounting-Request must contain the following attributes:

- Acct-Status-Type (40)
- Acct-Session-Id (44)
- Framed-IP-Address (8)

If any of these attributes is missing, the request is ignored.

Type of the request is defined by the Acct-Status-Type (40) attribute.

The following request types are recognized by UTM5 RADIUS:

Acct-Status-Type attribute value	Name	Comment
1	Start	Session start
2	Stop	Session end
3	Interim-Update	Intermediate data related to the established connection

- On receiving a Start packet:

    - An object describing the session is created and the UTM5 core is informed over Stream. The Acct-Session-Id (44) parameter contains the object ID.
    - If the login set in User-Name (1) belongs to some IP group or a dial-up service link, the last IP address of this IP group or a service link is marked busy for the time span defined by `radius_ippool_timeout`, if this parameter is set, or for three times as long as `interim_update_interval`, if this parameter is set (the former having higher priority).
    - IP address associated with the session is marked busy.
- On receiving a Stop packet:
    - If the login set in User-Name (1) corresponds to some IP traffic or hotspot service link, the session is tariffed based on the time set in Acct-Session-Time (46).
    - The object that describes session is removed.
    - IP address is marked as unused.
- On receiving an Interim-Update packet:
    - The object that describes session is modified.
    - If the login set in User-Name (1) belong to some dial-up service link, and the Interim-Update session control is on, the IP address of the given service link is marked busy for three times as long as `interim_update_interval`, if this parameter is set.

Tariffication by Stop packets

If the access server does not support the export of statistics by NetFlow, an option of tariffication by Stop packets may be used. For that the `radius_do_accounting` parameter must be set to 1.

On receiving the Stop packet, the RADIUS server creates two traffic records which later on are accounted in a standard way.

The first record contains the access server IP address for sender address, the subscriber IP address for destination and the Acct-Input-Octets (42) value from the Stop packet for the traffic amount consumed. The second record contains the NAS IP address for destination, the subscriber IP address for source, and the Acct-Output-Octets (43) value for the traffic amount.

The created records are sent to the UTM5 core over Stream.

Session control mechanism

If the `interim_update_interval` parameter (**Below**) is set, it is implied that the NAS sends Interim-Update packets periodically with this interval. When a packet does not arrive within three intervals, or when a Stop packet arrives, the session is dropped and the corresponding IP addresses released.

By default this parameter is not set, so the session may be dropped only upon arrival of Stop packets. If the NAS supports sending the Interim-Update packets, it is better to set this parameter to some reasonable value in order to avoid the occurrence of “hanging” sessions.

utm5_radius daemon

The UTM5 RADIUS executable file is called `/netup/utm5/bin/utm5_radius`.

Possible command line parameters are:

`-p <path>`	Path to the PID file
`-c <path>`	Path to the config file
`-V`	Version number and parameters information

The following options for `utm5_radius` startup are available:

1. Direct start of the `utm5_radius` executable with necessary parameters;
2. Start on watchdog with `start` parameter:

```
/netup/utm5/bin/safe_utm5_radius start
```

The script will restart `utm5_radius` automatically on failure;

1. Start via the automatic startup script (recommended).
   On Linux:

```
/etc/init.d/utm5_radius start
```

On FreeBSD or Solaris:

```
/usr/local/etc/rc.d/utm5_radius.sh start
```

To stop the `utm5_radius` and the watchdog script, execute:

on Linux:

```
/etc/init.d/utm5_radius stop
```

on FreeBSD or Solaris:

```
/usr/local/etc/rc.d/utm5_radius.sh stop
```

RADIUS server configuration

NetUP RADIUS server should be installed to `/netup/utm5/bin/utm5_radius`. Its parameters may be set up in the following ways:

- Via the config file;

- Via the administrator's interface (see **UTM5 core: Interface parameters** on page **152** for more detail).

  Config file parameters are used during the initialization of the RADIUS module.

Config file

RADIUS server running on Unix platforms uses `/netup/utm5/radius5.cfg` as its config file. For Win32 version, it is `radius5.cfg` at the installation directory (which by default is `C:\Program Files\NetUP\UTM5\`).

In order to set another configuration file, use `-c` command line parameter, e.g.

```
utm5_radius -c /etc/radius5.cfg
```

Config file has the following format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered a comment.

Below is the list of possible parameters:

Parameter	**Possible values**	**Default value**	**Description**
`core_host`	IP address	Mandatory parameter	Address of core host
`core_port`	1 – 65534	Mandatory parameter	Address of core port; disabled if `core_host` is not defined
`radius_login`	string	radius	System user login
`radius_password`	string	radius	System user password
radius_pid_file	file name	/var/ run/ utm5_radius.pid	PID file
radius_ping_in-terval	number	30	Maximum duration (in seconds) of repeated attempts to connect to the core
`radius_acct_host`	IP address	0.0.0.0	Host to accept Accounting-Request

Parameter	Possible values	Default value	Description
`radius_acct_port`	1 – 65534	1813	Port to accept Accounting-Request
`radius_auth_host`	IP address	0.0.0.0	Host to accept Access-Request
`radius_auth_port`	1 – 65534	1812	Port to accept Access-Request
`radius_auth_mppe`	enable	not set	Enables 128 bit MPPE on authorization via MS-CHAP-v2 protocol
`radius_auth_vap`	1	not set	If set, disables authorization of blocked users
`radius_ippool_acct_timeout`	time in seconds	30	Time in seconds for blocking IP addresses in the pool after sending Access-Accept packet
`radius_ippool_timeout`	time in seconds	not set	Time in seconds for blocking IP addresses in the pool after accepting the Accounting-Start packet. If not set, the addresses are blocked till accepting the Accounting-Stop packet. We do NOT recommend using this parameter
`radius_auth_null`	yes or enable	not set	If enabled, RADIUS server will accept and successfully authorize the requests without passwords when the user password is empty
`radius_auth_h323_remote_address`	enable, on, yes	not set	If enabled, the authorization is performed by Cisco VSA h323-remote-address attribute rather than by username attribute

Parameter	**Possible values**	**Default value**	**Description**
`radius_nas_port_vpn` a. Multiple instances of the parameter are possible.	integer number	not set (no checking)	If set, the NAS-Port-Type (61) attribute is checked against this value on authorization of the IP traffic user
`radius_nas_port_dialup` (a)	integer number	not set (no checking)	If set, the NAS-Port-Type (61) attribute is checked against this value on authorization of the dial-up user
`radius_nas_port_tel` (a)	integer number	not set (no checking)	If set, the NAS-Port-Type (61) attribute is checked against this value on authorization of the telephony user
`radius_nas_port_hotspot` (a)	integer number	not set (no checking)	If set, the NAS-Port-Type (61) attribute is checked against this value on authorization of the hotspot user
`radius_card_-autoadd`	yes, on, enable	not set (no registration)	Enables automatic registration of card users (card number and PIN stand for login and password, correspondingly).
`send_xpgk_ep_number`	any	not set (numbers not sent)	Enables sending the Cisco-AVPair (9;1) attribute having value `xpgk-ep-number=<` semicolon-separated list of numbers > in the Access-Accept request for telephony users

Parameter	Possible values	Default value	Description
`send_h323_ivr_in`	any	not set (numbers not sent)	Enables sending the Cisco-AVPair (9;1) attribute having value `h323-ivr-in=terminal-alias:<` semicolon-separated list of numbers > in the Access-Accept request for telephony users
`h323_origin_-reject`	string	not set	Sets zero cost for Accounting-Request having the h323-call-origin (9;26) attribute equal to this parameter's value
`interim_update_interval`	time in se-conds > 61	not set (standard mechanism used)	Enables advanced session control mechanism based on Interim-U-pdate packets. The value is passed via the Acct-Interim-In-terval (85) attribute of the Access-Accept packet
`radius_default_session_-timeout`	integer number	86400	Value of the Session-Timeout (27) attribute sent in Access-Accept for an IP traffic service
`radius_call-back_avpair_enable`	any	not set	Enables sending of the Cisco-AVPair (9;1) attribute having value `lcp:call-back-dialstring=<` callback n-umber >, where callback number is a part of login preceding the colon symbol

Parameter	**Possible values**	**Default value**	**Description**
`radius_acct_rewrite_login_answer`	enable, on, true	not set	Enables substitution of login with the h323-remote-address (9;23) attribute value for the Accounting-Request packets having the h323-call-origin (9;26) attribute set to `answer`
`radius_acct_rewrite_login_originate`	enable, on, true	not set	Enables substitution of login with the h323-remote-address (9;23) attribute value for the Accounting-Request packets having the h323-call-origin (9;26) attribute set to `originate`
`blocked_pool_name`	string	not set	Name of the IP pool to provide addresses for blocked users (in case those are entitled to some limited Internet access)
`guest_pool_name`	string	not set	Name of the IP pool to provide addresses for guest users (in case those are entitled to some limited Internet access)
`named_pool_shuffle`	yes, no	not set	Enables providing IP addresses from a random pool (if there are several with similar name). By default the addresses are issued from each pool in turn until it runs out; the pools follow in the order of addition

Parameter	Possible values	Default value	Description
radius_auth_tel_ext_reg	yes, on, enable	not set	Enables recognizing of registration request by the condition Calling-Station-Id = Called-Station-Id in Access-Request
tls_certificate_path	string	not set	Path to the certificate file when using EAP-TTLS
tls_pricate_key_path	string	not set	Path to the private key file when using EAP-TTLS
tel_session_timeout	integer number	86400	Maximum duration (in seconds) of a VoIP session
disconnect_request_timeout	integer number	5	PoD response timeout after manual drop of the session
incoming_trunk_format	any	not set	Incoming trunk format: vendor_id:attribute_id:regexp
outgoing_trunk_format	any	not set	Outgoing trunk format: vendor_id:attribute_id:regexp
pbx_id_format	any	not set	Call id format: vendor_id:attribute_id:regexp
override_service_type	true, false	false	Override service type in the incoming request. Set service type “framed”
dac_bind_host	IP address	0.0.0.0	Host to accept requests for connection parameters modification (incoming RADIUS server request)
h323_currency	string	USD	Currency code. Is used by IP telephony module
`use_closed_sessions_cache`	yes, on, enable	Not set	Store information about recently closed sessions in cache

Parameter	Possible values	Default value	Description
`closed_sessions_cache_size`	Integer number	0	Closed sessions cache size in number of sessions stored
h323_return_code_positive	yes, on, enable	not set	Make RADIUS server return Cisco:h323_return_code attributes with positive values.

Logging parameters (for more details see **System description: Logging** on page **22**):

Parameter	Possible values	Default value	Description
`log_level`	number from 0 to 3	1	Level of messages to be written to the log file
`log_file_main`	file name	standard error stream	Main log file
`log_file_debug`	file name	standard error stream	Debugging log file
`log_file_critical`	file name	standard error stream	Critical log file
`rotate_logs`	yes, on, enable	not set (rotation off)	Enables log files rotation
`max_logfile_size`	size in bytes	10485760	Maximum size of log file

a. Relevant if log files rotation is enabled.

Dynamic distribution of IP addresses

To enable tariffication of traffic in case of the dynamically assigned user’s IP addresses, a scheme of dynamic connection of IP address to service link is introduced.

The IP traffic service or dial-up/hotspot service link bound to the user’s account must have its **Dynamic IP addresses** option switched on.

On receiving the Accounting-Start request having login of the user in question for the User-Name (1) attribute and non-zero IP address for Framed-IP-Address (8), the UTM5 core sends an event to link this IP address to the account. The event contains the account details and the issued IP address. The event handler calls the function that performs all the necessary validations and the linking itself (if applicable).

Control flow of the said function goes as follows:

1. Search for the IP traffic service link having its **Dynamic IP addresses** option on and bound to the given account. If not found, skip the rest;
2. Search for the IP group to which the given IP address belongs. If found, remove the group;
3. Create an IP group with the given IP for address and 255.255.255.255 for mask (leave default values for the rest of parameters);
4. The IP group is connected to the service link found on step **1**.

Creation or removal of an IP group is associated with the following events: if the account owning the IP group in question has the Internet status **On**, it is switched to **Off** before the creation or removal of a group, and back to **On** after that.

It is strictly not recommended to use the dynamic IP addresses functionality if the address pool may overlap with the static IP addresses associated with IP traffic service links.

11

TEXT FILES IMPORT

Introduction

NetUP UTM5 supports importing text files containing the data on traffic and phone calls.

Some other entities may also be imported, albeit in a different way. For importing traffic subclasses from CSV files, see **Administrator’s interface: Traffic classes** on page **55**. For importing structured information related to some complex objects from XML files, see **Structured data import** on page **213**.

`utm5_send_traffic` should be employed to import traffic info in case if the said info contains neither sender nor destination address, but provides the data on traffic quantity, its class, and the login of the IP traffic service link to which the traffic belongs. If the option of providing traffic data via NetFlow is available, it should be used instead.

`utm5_send_cdr` should be used to import the info on phone calls in case if the provider of the said info does not support the RADIUS Accounting-Request. If the option of sending the phone calls info via the RADIUS Accounting-Requests is available, it should be used instead.

Prior to the version UTM5.3-001 both these tasks were performed by a single application called UTM5 Unif. Input files format and most of the config file parameters are retained fully compatible with those of UTM5 Unif.

Workflow

Parsing traffic info files

In case of parsing a traffic info file the following actions are performed:

1. A connection is established to the UTM5 core using URFA protocol.
2. The file is read line by line, and each line parsed according to standard format.
3. Data from each string are stored in the internal format.
4. Data structures in the internal format are passed to UTM5 by calling the URFA function `0x5511`.
5. `utm5_send_traffic` stops.

The traffic data file should contain the data in the following format:

1. Each string must be formatted like:

```
<LOGIN> <BYTES> <TCLASS> <IP>
```

where

- **LOGIN –** is the login of the IP traffic service link to which the traffic belongs;
- **BYTES –** is the traffic amount in bytes (should not exceed 2 GB);
- **TCLASS –** is the number of traffic class registered in UTM5 to which this traffic belongs;
- **IP –** is an IP address specified for this traffic in traffic reports grouped by IP. May contain arbitrary value.

1. The file should contain neither strings containing any other information, nor information in a different format.

Parsing phone call info files

In case of parsing a phone calls info file the following actions are performed:

1. A connection is established to the UTM5 core using URFA protocol.
2. The file is read line by line, and each line parsed according to standard format.
3. Data from each string are stored in the internal format.
4. Data structures in the internal format are passed to UTM5 by calling the URFA function `0x10310`.
5. `utm5_send_cdr` stops.

The phone calls data file should contain the data in the following format:

1. Each record of a phone call must be on a separate line.
2. No record may span more than one line.
3. Each record must conform to the format specified in the config file.
4. The file should contain neither strings containing any other information, nor information in a different format.

Each single record must meet the following requirements:

1. To contain text data on one call.
2. To contain several fields, including:
    - Calling party ID (telephone number);
    - Called party ID (telephone number);
    - Call length in seconds;
    - Call date and time, if the call is going to be recorded under date different from current;

The following optional fields may also appear:

- Incoming trunk;
- Outgoing trunk;
- PBX ID;
- Unique call ID (optional).

If the call date format is not specified in the config file, the following is assumed by default:

```
<hh>:<mm>:<ss>.<mil> <tzc> <dow> <mon> <dt> <yyyy>
```

where

Field	Length	Description
`hh`	2	Hours
`mm`	2	Minutes
`ss`	2	Seconds
`mil`	3	Milliseconds
`tzc`	3	Time zone code
`dow`	3	Day of week
`mon`	3	Month
`dt`	2	Date
`yyyy`	4	Year

For example, `00:35:05.000 UTC Tue Jul 19 2007`.

Milliseconds and day of week are ignored.

The field delimiter symbol is also specified in the config file and must be the same along the whole data file.

Utilities usage

The utilities are started as follows:

```
/netup/utm5/bin/utm5_send_traffic
```

and

```
/netup/utm5/bin/utm5_send_cdr
```

The acceptable command line parameters for both utilities are:

`-c <file>`	Path to the config file
`-s <file>`	Path to the data file to be imported. “ – ” denotes STDIN. By default, `/netup/utm5/source.dat` is used
`-v`	Version number and parameters information

Config files

The utilities use config files `utm5_send_traffic.cfg` and `utm5_send_cdr.cfg`, which on Unix platforms are located at `/net-up/utm5/`. In Win32 version they are placed into the installation directory (which by default is `C:\Program Files\NetUP\UTM5\`).

Config files have the following format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered a comment.

Below is the list of possible parameters.

Parameters of connection to the UTM5 core (present in both files):

Parameter	Possible values	Default value	Description
`core_host`	IP address	127.0.0.1	IP address of the UTM5 core
`core_port`	integer number from 1 to 65534	11758	Port of the UTM5 core that listens to URFA
`core_login`	string	init	System user login for UTM5
`core_password`	string	init	System user password

Parameters for parsing phone call info files (found only in `utm5_send_cdr.cfg`):

Parameter	**Possible values**	**Default value**	**Description**
`pbx_calling_sid`	integer number	0	Number of position containing the calling number
`pbx_called_sid`	integer number	1	Number of position containing the called number
`pbx_duration`	integer number	2	Number of position containing the call duration
`pbx_duration_format`	format string [a]	default format	Call duration format
`pbx_session_id`	integer number	3	Number of position containing the session ID
`pbx_date_time`	integer number	4	Number of position containing the date and time of the call
`pbx_date_format`	format string [b]	default format	Date and time format
`pbx_time`	integer number	not set	Number of position containing the time of the call (if stored separately from date)
`pbx_time_f-ormat`	format string [a]	not set	Time format
`pbx_accounting_code`	integer number	not set	Number of position containing the username (if it is included)
pbx_incoming_trunk	integer number	not set	Number of position containing the incoming trunk (if present)
pbx_outgoing_trunk	integer number	not set	Number of position containing the outgoing trunk (if present)
pbx_id	integer number	not set	Number of position containing the PBX ID (if present)
`pbx_-delimiter`	string	space	Field delimiter symbol
`pbx_quote`	string	empty string	Field enclosing symbol

a. Time format string may include specifiers %H, %h, %M, %m, %S, and %s, see below.
b. Date format string may include specifiers, see the full list below.

The following date and time format specifiers are available:

Specifier	**Description**
`%Y`	Four-digit year (1970...)
`%y`	Two-digit year (00..99)
`%N`	Month with leading zeros (01..12)
`%n`	Month without leading zeros (1..12)

Specifier	Description
`%H`	Hour with leading zeros (00..23)
`%h`	Hour without leading zeros (0..23)
`%D`	Day of the month with leading zeros (01..31)
`%d`	Day of the month without leading zeros (1..31)
`%M`	Minutes with leading zeros (00..59)
`%m`	Minutes without leading zeros (0..59)
`%S`	Seconds with leading zeros (00..60)
`%s`	Seconds without leading zeros (0..60)
`%b`	Three-letter month name (Jan..Dec)
%U	Time in unix timestamp format
`%z`	Time zone identifier (for example, GMT) – valid only for FreeBSD and Linux

Logging parameters (for more details see **System description: Logging** on page **22**):

Parameter	Possible values	Default value	Description
`log_level`	number from 0 to 3	1	Level of messages to be written to the log file (unless `-d` option is set)
`log_file_main`	file name	standard error stream	Main log file
`log_file_debug`	file name	standard error stream	Debugging log file
`log_file_critical`	file name	standard error stream	Critical log file

UTM5 RFW

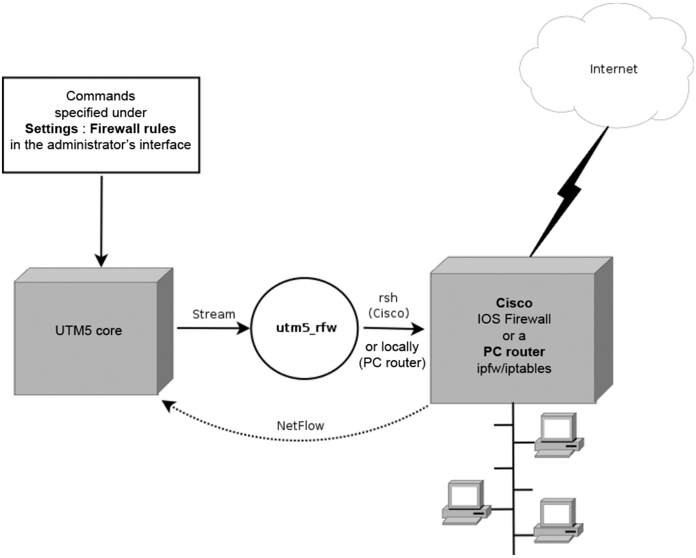
UTM5 RFW workflow

UTM5 RFW is a daemon that executes the commands issued by the UTM5 core. RFW is intended for controlling the external software, including firewalls, routers, shapers, etc.

In brief, UTM5 RFW works as follows:

1. Connect to the UTM5 core;
2. Receive commands from UTM5;
3. Execute commands locally or remotely.

On startup UTM5 RFW authorizes in the system using the parameters given in its config file. For successful authorization UTM5 RFW must be registered in the list of firewalls (see **Administrator's interface: Firewalls** on page **85**). The amount of UTM5 RFWs in the system is unlimited.



UTM5 RFW establishes permanent connection with the UTM5 core using the Stream protocol and awaits for commands issued by the core on some events. Commands for particular events are assigned in the administrator's interface on Settings: Firewall rules page.

The commands are executed either on the same server where RFW is running (if the firewall type is set to **Local**), either remotely over rsh, if it was set to **Remote Cisco**.

If the RFW is not connected to the core, the commands are cached for 24 hours. On reconnection of RFW to the core some commands may be executed according to the given synchronization parameters (see **Synchronization of rules** on page **196**); if the reconnection was without any flags, all cached rules will be sent to the affector. Also, an arbitrary command may be assigned for execution on RFW startup by the `firewall_flush_cmd` parameter (see **Config file** on page **194**).

Firewall rules

A firewall rule is an object that contains the command’s template and defines the conditions to execute the command.

The set of registered firewall rules may be found at Settings: Firewall rules page of the administrator’s interface.

Firewall rules have the following parameters:

- Applicability domain (to which users it is applicable);
- Initiating event (the event that causes the rule to execute);
- Place to apply (RFW to which the rule is passed and the corresponding firewall where it is actually executed);
- Template (the command proper).

The command templates may include variables which are substituted with their values on sending the command to RFW.

- **Applicability domain –** is a property that defines accounts to which the rule is applicable. If **All users** option is checked, the rule is applicable to all accounts in the system. Otherwise the subset of interest may be defined by user ID, by group, or by the tariff plan. Several conditions may be used simultaneously, combined by default with logical OR, though an alternative option of using logical AND is also available.

Selection by tariff plan covers the accounts having (among others) some tariff links with this tariff plan set as current. The rules applicable to service links or IP groups will be applied to all service links or IP groups related to these accounts, including those which by themselves are connected with different tariff plans.

- **Initiating event –** (one or more) is the event to trigger command execution is selected from the list (see **Events** on page **190**).
- **Place to apply –** selects the firewall to use from the list of existing firewalls, or probably selects all of them.
- **Command template –** is a string probably containing variables (see **Variables** on page **187**) which are substituted with their values on sending the command to the RFW.

When executing locally, an RFW calls the command as follows:

```
[sudo_path ][firewall_path ]arg1[ arg2[ arg3...]]
```

Here the optional parameters `sudo_path` and `firewall_path` are taken from the config file, and the rest is the command template where variables are already substituted with their values. Therefore the case when neither `sudo_path` nor `firewall_path` is set requires the command template to start with the name of some external executable file.

In UTM5 assembled for Win32 the `sudo_path` and `firewall_path` parameters are not used.

When executed over rsh, the command is sent as is, i.e. just the template with substituted variable values.

An example of firewall rules creation is described in **Creating firewall rules** on page **142**.

*Rules created in UTM5 versions prior to 5.2.1-007 must be converted (see **Rules conversion** on page **192**).*

Variables

Below is the list of all variables that may be used in command templates, together with their scope (i.e. the events to which every particular variable is applicable). If the variables are used outside of their scope, they are substituted with their default values.

Variable	Default value	Scope	Description
`UID`	empty string	All events excluding log file events	User ID
`UGROUP`	empty string		Semicolon-delimited list of IDs of groups to which the user belongs
`LOGIN`	empty string		User login
`EMAIL`	empty string	All events excluding log file events and **User deleted**	E-mail address set in the user's properties
`ACCOUNT_ID`	0		Account ID

Variable	**Default value**	**Scope**	**Description**
`RULE_ID`	0	All events excluding log file events and **User deleted**	Usre ID plus `fw_rule_offset` value from the list of system parameters
`FULL_NAME`	empty string		Full name of the user
`SWITCH_IP`	empty string		Firewall name set in the **Remote switch** field in the user’s properties (see User: Other on page 42)
`SWITCH_PORT`	empty string		**Port** set in the user’s properties (see User: Other on page 42)
`SLINK_ID`	0	All events excluding log file events, balance events, **User added**, and **User modified**	Service link ID
`ULOGIN`	empty string	Events of Internet, dial-up, hotspot, and D-ynashape	Login set in the properties of a service link or an IP group
`UIP`	`0.0.0.0`	Events of Internet, hotspot, and D-ynashape	User network address set in the IP group properties under **IP**
`UMASK`	`255.255.255.255`		Dot-separated network mask (for example, 255.255.255.0)
`UINVERTMASK`	`0.0.0.0`		Dot-separated inverted network mask (for example, 0.255.255.255). Used for Cisco routers
`UBITS`	`32`		Binary network mask (for example, 32 means 255.255.255.255)
`MAC`	empty string	Events of Internet and D-ynashape	MAC address set in the IP group properties
`SERVICE_ID`	0	Events of dial-up, session, IP traffic, telephony, IP-TV, and tech parameters	ID of a service
`UPASS`	empty string	Events of dial-up and hotspot	Password of a service link of dial-up or hotspot service

Variable	Default value	Scope	Description

`DIALUP_FLAGS`	empty string	Dial-up events	Flags of a dialup service link:
			0 – **Ringdown allowed** is set;
			3 – **Callback allowed** is set;
			1 – both are set

`UCID`	empty string		**CID** parameter value for a dial-up service link
`UCSID`	empty string		**CSID** parameter value for a dial-up service link
`DIALUP_LIST`	empty string	Blocking events	Semicolon-separated list of parameters of dial-up service links related to the given account in a form `"ID/ login/password/CID/CSID/flags"` for each link
`BLOCK_TYPE`	-1		Blocking type (see the full list in Accounts on page 24)
`SLINK_LIST`	empty string		Semicolon-separated list of service link IDs related to the given account
BALANCE	0	Balance events	Account balance
`TECH_PARAM_TYPE`	0	Tech parameter events	Type of the tech parameter (1 stands for web, 2 for e-mail)
`TECH_PARAM_ ID`	0		Tech parameter ID
`TECH_PARAM_VALUE`	empty string		Tech parameter value
`TECH_PARAM_PASS`	empty string		Tech parameter password
`SERVICE_TYPE`		Events of session and tech parameters	Type of service (see the full list in Services on page 28)
`TARIFF_LINK_ID`	0	IP traffic events	Tariff link ID
`DISCOUNT_PERIOD_ID`	0		Accounting period ID
`START_DATE`	0		Staring date of a service link
`END_DATE`	0		Ending date of a service link
`IP_GROUP_ID`	0		IP group ID of a service link
IPTV_ SERVICE_ DATA	empty string	IPTV events	The contents of Custom options field in IPTV service parameters
`IP_GROUP_LIST`	empty string	Events of blocking and IP traffic	Semicolon-separated list of IP groups in a form “ `address/mask/ login/password/MAC/ NetFlow provider` ” for each group

Variable	Default value	Scope	Description
`TIME_LIMIT`	0	Hotspot events	Remaining time of service for hotspot
TEL_LIST	empty string	Telephony events and Blocking events	Semicolon-separated list of telephone numbers in a form “`number/login/CIDs`” for each number
`NAS_ID`	empty string	Session events	ID of a NAS
`NAS_IP`	0.0.0.0		IP address of a NAS
`SESSION_ID`	empty string		ID of a session
`ACCT_STATUS_TYPE`	0		Session status (1 for open, 2 for closed)
`CALLING_SID`	empty string		ID of the calling station
`CALLED_SID`	empty string		ID of the called station
`FRAMED_IP`	0.0.0.0		IP address set in the Framed-IP-Address (8) RADIUS attribute
`BANDWIDTH`	0	Dynashape events	Currently permitted bandwidth
`PATH`	empty string	Log file events	Path to the log file or statistics file

The `SPLINK_ID`, `TRAFFIC_LIMIT`, `UTELLOGINS`, `UTELNUMBERS`, and IP_LIST variables are deprecated and out of use.

The following enumerations may be used in some variables:

- Blocking types (see **Accounts** on page **26**);
- Service types (see **Services** on page **31**).

Events

Below is the list of events that may trigger a command:

Internet events

- **Internet on –** executes for each IP group in every IP traffic service link related to the given account when the Internet status for this account is changed to **On**;
- **Internet off –** executes *twice* for each IP group in every IP traffic service link related to the given account when the Internet status for this account is changed to **Off**;

User events

- **User added –** executes for the user being added to the system via the administrator’s interface or automatically;
- **User modified –** executes for the user whose data has been changed;

- **User deleted –** executes for the user which is being deleted;

Blocking events

- **Block type changed –** executes for an account on changing its blocking state (that is, on blocking or unblocking);

Balance events

- **Balance notification sent –** executes for an account when its balance passes by the threshold defined by the system parameter `notification_borders`;

Session events

- **Session opened –** executes for a service link on Accounting-Start RADIUS request;
- **Session closed –** executes for a service link on Accounting-Stop RADIUS request;

Dialup events

- **Dialup link added –** executes for a dial-up service link on its creation;
- **Dialup link modified –** executes for a dial-up service link on changing its parameters;
- **Dialup link deleted –** executes for a dial-up service link on its removal;

IP traffic events

- **IP traffic link added –** executes for an IP traffic service link on its creation;
- **IP traffic link modified –** executes for an IP traffic service link on changing its parameters;
- **IP traffic link deleted –** executes for an IP traffic service link on its removal;

Telephony events

- **Telephony service link added –** executes for a telephony service link on its creation;
- **Telephony service link modified –** executes for a telephony service link on changing its parameters;
- **Telephony service link deleted –** executes for a telephony service link on its removal;

Hotspot events

- **Hotspot enabled –** executes for a hotspot service link on user’s authorization;
- **Hotspot disabled –** executes for a hotspot service link on user’s logout or session stopping;

Tech parameters events

- **Tech parameter added –** executes for a service link on creation of a technical parameter (see Tariffication: Technical parameters on page 43) related to it;
- **Tech parameter modified –** executes for a service link on changing a technical parameter related to it;
- **Tech parameter deleted –** executes for a service link on removal of a technical parameter related to it;

Dynashape events

- **Set bandwidth limit (incoming) –** executes for each IP group on approaching the shaping conditions (see **Administrator’s interface: Dynamic shaping** on page **96**) imposed on the given service link for the incoming channel;
- **Edit bandwidth limit (incoming) –** executes for each IP group on changing the shaping conditions imposed on the given service link for the incoming channel (say, when the amount of traffic passes over the border limits);
- **Delete bandwidth limit (incoming) –** executes for each IP group on leaving the shaping conditions imposed on the given service link for the incoming channel (say, when the amount of traffic is zeroed at the end of accounting period);
- **Set bandwidth limit (outgoing) –** executes for each IP group on approaching the shaping conditions imposed on the given service link for the outgoing channel (say, when the amount of traffic reaches the lower border limit);
- **Edit bandwidth limit (outgoing) –** executes for each IP group on changing the shaping conditions imposed on the given service link for the outgoing channel;
- **Delete bandwidth limit (outgoing) –** executes for each IP group on leaving the shaping conditions imposed on the given service link for the outgoing channel;

Log file events

- **Raw traffic file closed –** executes for the detailed statistics file on its closing;
- **Log file closed –** executes for the log file on its closing;

DHCP lease events

- **New DHCP lease –** executes on new IP address allocation (DHCP lease is offered);
- **DHCP lease update –** executes when a DHCP lease is updated;
- **DHCP lease expire –** executes when a DHCP lease expires;

IPTV events

- **IPTV link added –** executes on IPTV service link creation;
- **IPTV link modified –** executes on IPTV service link modification;
- **IPTV link deleted –** executes on IPTV service link deletion.

Rules conversion

Firewall rules created in UTM5 version 5.2.1-006 or earlier must be converted using the `fix_fwrules` program. Possible command lines parameters are:

`-f`	Convert rules
`-c <path>`	Path to the UTM5 config file, by default `/netup/utm5/utm5.cfg`, in Win32 version – `C:\Program Files\NetUP\UTM5\utm5.cfg`).
`-l <path>`	Path to the log file, by default `./fix_fwrules.log`
`-h`	Version number and parameters information

Thus, when UTM5 is installed to the default path, the conversion is started as follows:

```
fix_fwrules -f
```

For Win32 version it is:

```
C:\Program Files\NetUP\UTM5\bin\fix_fwrules -f
```

Firewall

Firewall is a system object used to identify an affector, a commutator, or a NetFlow provider.

The Settings: Firewalls page of the administrator's interface lists the registered firewalls and contains the interface for adding, modifying or deleting them.

A firewall is characterized by the following parameters:

- **ID –** is assigned automatically.
- **Type –** is **Local** to execute the commands locally, or **Remotte Cisco** to execute them remotely over rsh. Must conform to the `firewall_type` parameter set in the RFW config file of this firewall.
- **Name –** is a unique name to identify the RFW. Must conform to the `rfw_name` parameter set in the RFW config file of this firewall.
- **IP –** is an IP address of NetFlow provider to be set in the properties of an IP group.
- **Login –** is a login to use as `remote login` in rsh authorization. Relevant only for the **Remote Cisco** type. The `local login` is always set to `netup`.
- **Comment –** is an arbitrary comment.

An example of firewall creation is described in the **Creating firewall rules** on page **142**.

utm5_rfw settings

The `utm5_rfw` executable file is called `/netup/utm5/bin/utm5_rfw` .

In the Win32 version of UTM5 the file is called `utm5_rfw.exe` *and is located in the* `bin` *subdirectory of the installation directory (by default* `C:\Program Files\NetUP\UTM5\`*).*

Possible command line parameters are:

`-c <cfg>`	Config file path
`-s <flags>`	Synchronize firewall rules on startup. Possible flags are listed at **Synchronization of rules** on page **196**
`-f`	Deprecated, superseded with `-s enable`
`-o`	Deprecated, superseded with `-s disable`
`-v`	Version number and parameter information

In the Win32 version of UTM5 RFW runs as a system service, so the command line parameters can not be used. The `sync_flags` config file parameter may serve as a substitute for `-s`.

The following options for `utm5_rfw` startup on unix systems are available:

1. Direct start of the `utm5_rfw` executable with necessary parameters;
2. Start on watchdog with `start` parameter:

```
/netup/utm5/bin/safe_utm5_rfw start
```

In this case the `-f` command line parameter is effectively passed to the executable. The script will restart `utm5_rfw` automatically on failure;

1. Start via the automatic startup script (recommended).
On Linux:

```
/etc/init.d/utm5_rfw start
```

On FreeBSD or Solaris:

```
/usr/local/etc/rc.d/utm5_rfw.sh start
```

To stop the `utm5_rfw` and the watchdog script, execute:

on Linux –

```
/etc/init.d/utm5_rfw stop
```

on FreeBSD or Solaris –

```
/usr/local/etc/rc.d/utm5_rfw.sh stop
```

To start an RFW on a remote machine, it is essential that its config file parameters `core_host` and `core_port` conform to the address and port used by the UTM5 core for Stream protocol connections.

Several RFWs may run on the same machine simultaneously, given that they have separate config and PID files.

Config file

By default, UTM5 RFW on unix systems uses the config file `/netup/utm5/rfw5.cfg`. The Win32 version of UTM5 uses `rfw5.cfg` located in the installation directory (by default `C:\Program Files\NetUP\UTM5\`).

Config file has the following format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered a comment.

Below is the list of possible parameters.

Parameter	**Possible values**	**Default value**	**Description**
Common parameters:			
`rfw_name`	string	Mandatory parameter	Name of the UTM5 RFW by which it is known to the UTM5 core. Must conform to the firewall's **Name** parameter
`core_host`	IP address	Mandatory parameter	IP address of the host where the UTM5 core is running
`core_port`	number from 1 to 65534	Mandatory parameter	Port where the UTM5 core listens to Stream
`rfw_login`	string	Mandatory parameter	System user login to UTM5
`rfw_ password`	string	Mandatory parameter	System user password to UTM5
`firewall_ type`	local, cisco	local	Firewall type. Must conform to the **Type** parameter of the corresponding firewall
`rfw_ssl_ type`	tls1, ssl3, none	ssl3	SSL secure connection type (non-encrypted if set to none)
Relevant for `firewall_type=local`:			
`sudo_path`	executable file name	not set	Name of the `sudo` executable file
`firewall_path`	executable file name	empty string	Name of the executable file controlling the external software
`firewall_flush_cmd`	executable file name	empty string	Script running on connection and reconnection to the core
`dont_fork`	yes, enable, true	not set (execute in parallel)	Enables sequential execution, so that each rule will be run after the previous one is completed. Recommended to use along with `ipt-ables`
Relevant for `firewall_type=cisco`:			
`cisco_ip`	IP address	Mandatory parameter	IP address to send the rsh commands to.

Logging parameters (for more details see **System description: Logging** on page **22**):

Parameter	**Possible values**	**Default value**	**Description**
`log_level`	number from 0 to 3	1	Level of messages to be written to the log file
`log_file_main`	file name	standard error stream	Main log file
`log_file_debug`	file name	standard error stream	Debugging log file
`log_file_critical`	file name	standard error stream	Critical log file
`rotate_logs`	yes, on, enable	not set (rotation off)	Enables log files rotation
`max_logfile_count` a. Works if log files rotation is enabled.	number	not set (unlimited)	Maximum number of log files to keep
`max_logfile_size` (a)	size in bytes	10485760	Maximum size of log file
`pid_file`	file name	/var/run/utm5_rfw.pid	PID file
syslog_name	string	Not set	Log entry prefix (when logging to syslog is enabled)

The `core_timeout` parameter is deprecated and out of use.

Synchronization of rules

On automatic startup of RFW some rules may be executed in order to reinstall the configuration of the external software. Only the rules related to the given RFW may be executed, and only on this RFW itself. The set of rules to execute is defined by the flags (listed below).

The flags may be passed either via the config file parameter `sync_flags` or via the command line parameter `-s`, the latter having higher priority. When several flags used simultaneously, they must be joined into a colon-separated string.

Available flags include:

- `enable` executes the rules related to the **Internet on** event;
- `disable` executes **Internet off**;
- `users` executes **User added** for all users;
- `iptraffic` executes **IP traffic link added** for all such links;
- `dialup` executes **Dialup link added** for all such links;
- `blocks` executes **Block type changed** for all accounts;
- `shaping` executes **Set bandwidth limit** for all IP groups.

13

UTM5 DYNASHAPE

Introduction

UTM5 Dynashape module is intended to impose limitations on the user's channel bandwidth. The bandwidth limit may depend on time and on the amount of traffic consumed by the user.

The UTM5 system core composes the firewall rules according to the settings, and passes them for execution to the external software. The development of shaping control executables for the particular networking configuration to the administrator's area of responsibility. See the examples in **Appendix: Approaches to traffic shaping** on page **289**.

Dynashape module requires a separate license. To verify the availability of the license and its term of validity, see **About: Licenses** in the UTM5 administrator's interface and check the list item denoted **DynaShape module**.

Workflow description

Shaping may be set up separately for each IP traffic service. The shaping settings include the collection of bandwidth limitations and the associated parameters that determine the conditions to apply each limitation.

To set up shaping for a service:

1. Select the service ans set up the basic shaping parameters (see Administrator's interface: Dynamic shaping on page 86), including:
   - types of IP groups to which the shaping applies (VPN or non-VPN);
   - time range(s) when the shaping applies;
   - border values of traffic amount, on passing which the limitations will be applied sequentially;
   - bandwidth values for each time range and for each border value;
   - traffic classes to which the shaping applies.
1. In case of shaping with the RADIUS attributes, provide these attributes on the same page. The dynamic adjustment of attributes depending on the allowed bandwidth is enabled by the use of variables (see **RADIUS parameters**).
2. Set the firewall rules (see **Administrator's interface: Firewall rules** on page **86**) for the events **Set bandwidth limit**, **Edit bandwidth limit** and **Delete bandwidth limit** for incoming and outgoing traffic, using the `BANDWIDTH` variable which on application is substituted with the allowed bandwidth value.

The limitations are applied during the selected time range(s) to the IP groups of selected kind(s) and to the selected traffic classes, according to the amount of traffic consumed by the given service link. On getting under the shaping conditions, or on changing those (i.e. on starting the time range for which the shaping is set, or when the traffic amount surpasses the given border), correspondingly, the **Set bandwidth limit** or **Edit bandwidth limit** event for incoming or outgoing traffic occurs. In addition, the respective RADIUS attributes are sent and their previous values (if any) removed. On running away from the shaping conditions (i.e. when the corresponding time range finishes, or when the traffic amount is nullified at the end of accounting period) the **Delete bandwidth limit** event is executed and the RADIUS attributes deleted.

When a new time range starts, the ensuing limitations are set within 5 minutes from the beginning of the said range. The limitations due to consumed traffic amount are set after the next aggregation, which happens at regular intervals defined by the `traffic_agregation_interval` parameter (see **UTM5 core: Interface parameters** on page **152**).

In case of shaping by external scripts the UTM5 core passes to them the given bandwidth value as is. On the contrary, in case of RADIUS attributes-driven shaping the input value is interpreted as Kbits/sec, and may be converted to other units as well as to derivative values, see **RADIUS parameters**.

RADIUS parameters

The RADIUS attributes may include variables which get replaced with the respective values calculated from the bandwidth at given conditions. Below is the list of available variables:

Variable	**Description**	**Value (here W is the given bandwidth)**
`IN_BANDWIDTH_BITS`	Incoming bandwidth in bits/sec	W*1024
`IN_BANDWIDTH_KBITS`	Same, in Kbits/sec	W
`IN_BANDWIDTH_MBITS`	Same, in Mbits/sec	W/1024
`OUT_BANDWIDTH_BITS`	Outgoing bandwidth in bits/sec	W*1024
`OUT_BANDWIDTH_KBITS`	Same, in Kbits/sec	W
`OUT_BANDWIDTH_MBITS`	Same, in Mbits/sec	W/1024
`IN_CISCO_NORMAL_BURST`	Incoming burst size in bytes	1.5*(W*1024)/8
`IN_CISCO_EXTENDED_BURST`	Incoming extended burst size in bytes	1.5*2(W*1024)/8
`OUT_CISCO_NORMAL_BURST`	Outgoing burst size in bytes	1.5*(W*1024)/8

Variable	Description	Value (here W is the given bandwidth)
`OUT_CISCO_EXTENDED_BURST`	Outgoing extended burst size in bytes	1.5*2*(W*1024)/8

14

UTM5 URFACLIENT

Warning

The UTM5 urfaclient module provides direct interface for low-level operations which (unlike those performed via the control center or the web interface) may lack proper verifications or the coupled actions necessary to maintain the data integrity. Therefore any urfaclient operation to be applied to the production system must be thoroughly checked in the test environment beforehand.

NetUP does not assume responsibility for any possible losses caused by incorrect usage of the urfaclient module.

Introduction

The UTM5 urfaclient module is intended for the unified access to the UTM5 core data structures via RPC interface (URFA).

UTM5 urfaclient is composed of the following parts:

- core library `liburfa-client.so` that provides the necessary means for the interaction of the `utm5_urfaclient` utility and UTM5 core;
- `utm5_urfaclient` utility that actually performs the requested actions;
- schemes describing input and output parameters of the involved URFA functions;
- specific URFA scripts for any particular action or a sequence thereof.

Output of the executed URFA functions is directed to stdout.

The UTM5 urfaclient module requires a separate license. To verify the availability of the license, see About: Licenses in the UTM5 administrator interface and check the list item denoted **URFA client**.

Scheme

The `api.xml` scheme contains XML description of the following:

- input and output parameters of various functions;
- action sequence, probably dependent on the parameters' values.

Path to `api.xml` may be passed via the command line key `-api`. By default, it is `/netup/utm5/xml/api.xml`.

Expression value is either the value of a variable, or the output value of a built-in function, or a constant, if neither a variable nor a function with such name exist.

Variables are actually arrays of strings. When array index is not specified, zeroth element is assumed by default.

Interpretation of variables is context-dependent. Say, an `integer` tag implies parsing of string on return and serialization on assignment.

All variables belong to the global scope, so care must be taken to avoid name conflict.

The built-in system functions are:

- `now()` returns string representation of current time in unix format;
- `max_time()` returns string representation of maximum possible time in UTM5 in unix format (2000000000, year 2033);
- `size(varname)` returns the length of the `var_name` array.

Tags available

- **urfa –** is a root tag. Has no attributes. May contain one or several `function` tags.
- **function –** describes a function. Mandatory attributes:
  - `name`, the function name;
  - `id`, the function ID.

  Mandatory tags are: `input` and `output` (one for each function’s description) in arbitrary sequence.
- **input –** contains description of the function’s input parameters. Has no attributes. May contain an ordered sequence of the following tags:
  - `integer`
  - `long`
  - `double`
  - `string`
  - `ip_address`
  - `if`
  - `for`
  - `error`
- **output –** is the same as `input`, only for output parameters of a function.
- **integer –** may reside either in `input` or in `output`. Contains 32-bit signed integer (`int32_t`). Must have the `name` attribute with variable name. May also have attributes:
  - `default` – default value. Relevant only for input parameters, in case if the corresponding variable has not been found. If both the variable and the default value are absent, the program will abort and return a non-zero error code.
  - `array_index` – source or destination array index.

- **long –** is the same as `integer`, only for 64-bit signed integer (`int64_t`).
- **double –** is the same as `integer`, only for floating point number (`double`).
- **string –** is the same as `integer`, only for string parameters.
- **ip_address –** is the same as `integer`, only for IPv4 address (for example, 192.168.0.1 or 255.255.0.0) internally represented as `int32_t`.
- **if –** provides conditional operator in a sequence of parameters depending on the variable value. Must have the following attributes:
  - `variable`, which is the name of the variable to be checked;
  - `value` to check against;
  - `condition` to check (`eq` for “equal”, `ne` for “not equal”).

  May contain an ordered sequence of the following tags:
  - `integer`
  - `long`
  - `double`
  - `string`
  - `ip_address`
  - `if`
  - `for`
  - `error`

  Other nested tags are not allowed.
- **for –** provides loop operation. Must have the following attributes:
  - `name` of the iterator variable;
  - `from` (starting iterator value);
  - `count` (number of iterations).

  May contain an ordered sequence of the following tags:
  - `integer`
  - `long`
  - `double`
  - `string`
  - `ip_address`
  - `if`
  - `for`
  - `error`

  Other nested tags are not allowed.
- **error –** causes exit with a non-zero error code. May have the following attributes:
  - `icode` (exit code to return);
  - `comment` (error description);
  - `variable` to print out after the comment.

URFA scripts

URFA script describes a sequence of URFA function calls, loops and conditional operators in a form of XML tags.

Name of the directory containing URFA scripts may be passed via the command line key `-x`. The default value is `/netup/utm5/xml/`. Each action correspond to one file called `<action_name>.õml`. For example, the `add_user` action by default uses the file `/netup/utm5/xml/add_user.xml`.

URFA script must conform to the scheme.

Tags available

- **urfa –** is a root tag. Must contain an ordered sequence of the following tags:
  - `call`
  - `parameter`
  - add
  - sub
  - mul
  - div
  - cat
  - `if`
  - `for`
  - `message`
  - out
  - `set`
  - `error`
  - `remove`
- **call –** performs an URFA function call.
  Must have the `function` attribute (name of the function to call).
  May have the `output` attribute (if set to zero, XML output is blocked).
  May contain `parameter` tags.
  Other nested tags are not allowed.
- **parameter –** describes a parameter. Must have the `name` attribute (variable name).
  May provide input value if contains the `value` attribute; otherwise defines a command line parameter.
  If a variable is set both via the `value` attribute in the action file and via the command line,

the latter has higher priority. All values of the given parameter or its default value are placed in an array with name defined in the `name` attribute.

Multidimensional arrays, whenever required, must be entered by means of the data file (see ***Data files*** *on page* ***209****).*

Also may have the `comment` attribute containing the description of the variable which may be printed out via combined use of the command line keys `-a [action_name]` and `-help`.

- **add –** is the tag of arithmetic addition. Must have attributes:
  - `arg1` – first addend,
  - `arg2` – second addend,
  - `dst` – name of the variable to store the result.
- **sub –** is the tag of arithmetic subtraction. Must have attributes:
  - `arg1` – minuend,
  - `arg2` – subtrahend,
  - `dst` – name of the variable to store the result.
- **mul –** is the tag of arithmetic multiplication. Must have attributes:
  - `arg1` – first multiplicand,
  - `arg2` – second multiplicand,
  - `dst` – name of the variable to store the result.
- **div –** is the tag of arithmetic division. Must have attributes:
  - `arg1` – dividend,
  - `arg2` – divisor,
  - `dst` – name of the variable to store the result.
- **cat –** is the tag of string concatenation. Must have attributes:
  - `arg1` – first string,
  - `arg2` – second string,
  - `dst` – name of the variable to store the result.
- **if –** provides conditional operator in a sequence of parameters depending on the variable value. Must have the following attributes:
  - `variable`, which is the name of the variable to be checked;
  - `value` to check against;
  - `condition` to check (`eq` for “equal”, `ne` for “not equal”).

  May contain an ordered sequence of the following tags:
  - `call`
  - `parameter`
  - add
  - sub

  - mul
  - div
  - cat
  - `if`
  - `for`
  - `message`
  - out
  - `set`
  - `error`
  - `break`
  - `remove`

  Other nested tags are not allowed.

- **for –** provides loop operation. Must have the following attributes:
  - `name` of the iterator variable;
  - `from` (starting iterator value);
  - `count` (number of iterations).

  May contain an ordered sequence of the following tags:
  - `call`
  - `parameter`
  - add
  - sub
  - mul
  - div
  - cat
  - `if`
  - `for`
  - `message`
  - out
  - `set`
  - `error`
  - `break`
  - `remove`

  Other nested tags are not allowed.
- **message –** must have the `text` attribute. Outputs a debugging message to `STDOUT`.
- **out –** must have the `var` attribute. Outputs a variable defined by `var` to `STDOUT`.
- **set –** sets the variable value. Must have the following attributes: `dst` and either `src` or `value`. Simultaneous use of `src` and `value` is forbidden.
  May also have the following attributes:

  - `dst` defines the name of the destination variable (created if does not exist);
  - `src` defines the name of the source variable;
  - `dstindex` is the array index for destination (0 assumed by default);
  - `srcindex` is the array index for source (0 assumed by default);
  - `value` is the expression to assign to the variable.

  New element of an array may be written either to an existing element or to the next adjacent one (i.e. the element with index equal to the current size of the array). Indexes start from 0. Calls to out-of-range elements cause program abort.

- **error –** causes exit with a non-zero error code. May have the following attributes:
  - `icode`, the exit code to return;
  - `comment` (error description);
  - `variable` to be printed out after the comment.
- **shift –** shifts the `name` array one step to the left, removing the first element. Usage not recommended.
- **break –** interrupts the innermost `for` tag execution and continues from the following line.
- **remove –** removes the whole array `name`, or its single element with index `arrayindex`, in case if the `arrayindex` attribute is given. In this case the subsequent elements are shifted left by one.

Data files

Data file describes an array or several arrays of data to be used as input parameter(s) in a function. Data file should be passed to urfaclient via the command line key `-datafile`.

Tags available

- **urfa –** is a root tag. Must contain a sequence of `array` tags.
- **array –** is a top-level array.
  Mandatory attributes:
  - `name` (variable name).
  - `dimension` (array dimension).

  Also may have the `comment` attribute containing an arbitrary comment.
  Must contain an ordered sequence of `dim` tags.
  Other nested tags are not allowed.
- **dim –** describes an array element, which itself may or may not be an array.
  May have the `comment` attribute containing an arbitrary comment.
  Must contain either a value or an ordered sequence of `dim` tags.
  Other nested tags are not allowed.

utm5_urfaclient utility

UTM5 urfaclient is called as follows:

```
/netup/utm5/bin/utm5_urfaclient [parameters]
```

Each command line parameter consists of the space-separated key-value pair, with the exception of the `-help`, `-debug`, `-u`, and `-dealer` keys which require no value.

Most of parameters have their counterparts in the config file. The complete list of command line keys and config file parameters is given below.

Besides that, some action-specific parameters are possible. Depending on the nature of the action, they may or may not be mandatory.

All string values must be passed in UTF-8 encoding.

Order of parameters is not important.

Config file settings

UTM5 Urfaclient uses the config file `/n-etup/utm5/utm5_urfaclient.cfg` on Unix platforms. For Win32 version, it is `utm5_urfaclient.cfg` at the installation directory (which by default is `C:\Program Files\NetUP\UTM5\`).

Config file has the following format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered a comment.

All parameters may be passed to the program via the command line as well. The command line parameters have priority over those given in the config file and in the data file (if any).

The list of available parameters and command line keys is given below.

Key	Parameter	Default value	Description
`-h`	`core_host`	127.0.0.1	IP address of the host where UTM5 is running
`-p`	`core_port`	11758	Port of the aforementioned host listening to URFA
`-l`	`core_login`	init	Login for access to the UTM5 core
`-P`	`core_password`	init	Password for access to the UTM5 core
`-x`	`xml_path`	/netup/utm5/xml/	Path to the scheme and URFA scripts (a)
`-api`	api	/netup/utm5/xml/api.xml	Path to the scheme (if different from `xml_path`) (a)
`-u`	`plain_user`	not set	If set to "`yes`", makes urfaclient log in as plain user. In this case, only the user functions may be called (i.e. those with negative IDs); otherwise it is vice versa
`-dealer`	dealer	not set	If set to "`yes`", only dealer functions may be called
`-s`	`session_key`	not set	Enables persistent sessions. Login and password must be provided either in the config file or in the command line parameters
`-i`	`user_ip`	127.0.0.1	When restoring a persistent session, denotes the IP address from which the session has been established initially
`-a`	n/a	not set	Action name (mandatory)
-c	n/a	/netup/utm5/utm5_urfaclient.cfg	Path to the config file (b)
`-help`	n/a	not set	Outputs help info. When used together with `-a`, produces help on particular action (if available)
`-debug`	n/a	not set	Enables additional debugging output, including the inner variables' values
`-datafile`	n/a	not set	Path to the data file
-<name>	n/a	not set	Value of the function input parameter called <name>

a. In UTM5 assembled for Win32 by default the scheme file and scripts are located at `C:\Program Files\NetUP\UTM5\share\`.

b. On Win32 it is `C:\Program Files\NetUP\UTM5\utm5_urfaclient.cfg`.

Usage example

Sample URFA scripts and other files are located at `/netup/utm5/xml` (in UTM5 assembled for Win32 by default it is `C:\Program Files\NetUP\UTM5\share`). Of all XML files found there, `api.xml` is the scheme file, `search_users_new_data.xml` and `teldata.xml` are data files, and the rest are scripts.

The example contains tariff plan assignment for a user, together with attachment of all services listed in the plan as **Attach by default**.

Below is an example call of urfaclient utility with parameters:

```
utm5_urfaclient -a link_tariff_with_services -user_id 5
   -account_id 5 -discount_period_id 2 -tariff_current 1
   -ip_address 10.4.5.7 -iptraffic_login test4
   -iptraffic_password 123
```

In this example a `link_tariff_with_services.xml` script is called, which attaches certain tariff plan to a specified user account, together with attachment of all services having their **Attach by default** flag set.

The resulting output is directed to STDOUT.

15

STRUCTURED DATA IMPORT

Introduction

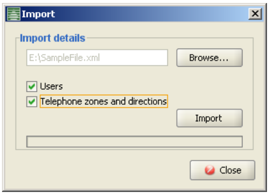
NetUP UTM5 supports data import in a form of XML files containing the following entities: users, telephone directions and telephone zones.

Some other entities may also be imported, albeit in a different way. For importing traffic subclasses from CSV files, see **Administrator's interface: Traffic classes** on page **55**. For importing traffic and phone call data from text files, see **Text files import** on page **179**.

Interface

To import data:

1. From the top menu of the control center, select **Menu: Import**.
2. Press **Browse** and browse to the XML file.
3. Set the check boxes corresponding to the entities you want to import. (Other entities contained in the file, if any, will be ignored.)
4. Press **Import**.



E:\SampleFile.xml

☑ Users

☑ Telephone zones and directions

The imported XML file is checked for consistency to the scheme (see below). Non-redundancy of entries and validity of internal cross-references (i.e. the existence of entities referenced in the data being imported) are also checked. On success the file is imported into the database. Otherwise, an error message pops up and the import operation does not occur.

XML file scheme

The full XML scheme is located at **http://www.netup.ru/xsd/import.xsd**. An example XML file is presented below in **Example XML file** on page **219**.

The XML tree of the file begins with the top-level `import` element containing `users`, `zones`, and `directions` nodes, which in turn may include an arbitrary number of child nodes `user`, `zone`, and `direction`, correspondingly (see the descriptions below).

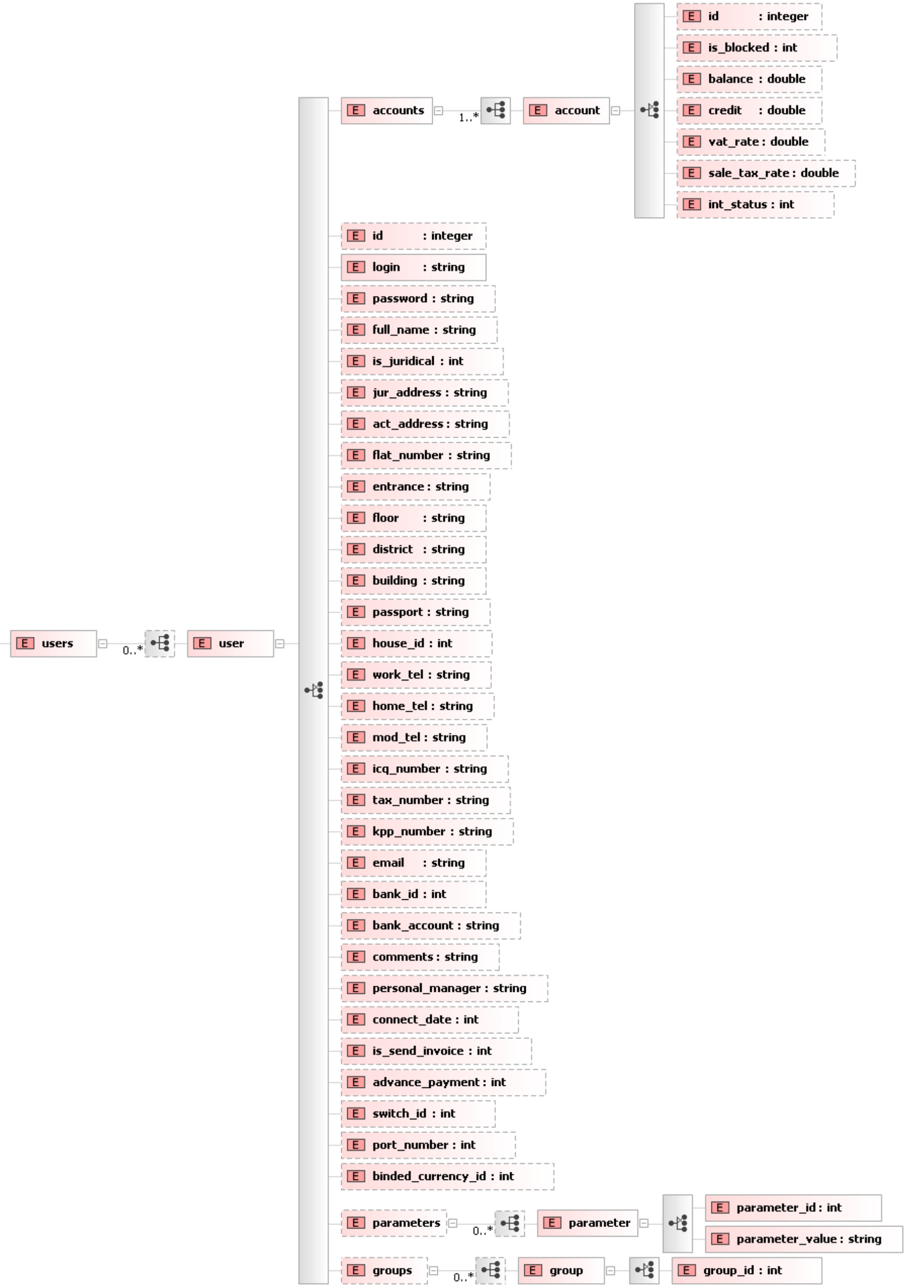


Рис. 1. XML file scheme (`users` element).

Elements: user

Contains description of a user. May include the following elements (login and one account are mandatory, the rest is optional):

Element	Type	Default value	Description
`login`	string	Mandatory element	User login
`accounts`	Collection of `account` elements (see **account** on page **216**). Must be present and contain at least one element		
`id`	number	Not set	User ID (reserved for future use). Not related to the actual ID assigned to the user when recorded into the database
`password`	string	Not set	Password
`full_name`	string	Not set	Full name
`is_juridical`	0, 1	0	0 for an individual, 1 for a legal entity
`jur_address`	string	Not set	Legal address
`act_address`	string	Not set	Actual address
`district`	string	Not set	District
`building`	string	Not set	Building
`entrance`	string	Not set	Entrance
`floor`	string	Not set	Floor
`flat_number`	string	Not set	Flat number
`passport`	string	Not set	Passport data
`house_id`	number	Not set	ID of the house in UTM
`work_tel`	string	Not set	Work phone
`home_tel`	string	Not set	Home phone
`mod_tel`	string	Not set	Mobile phone
`icq_number`	string	Not set	ICQ number
`tax_number`	string	Not set	Tax payer identification number
`kpp_number`	string	Not set	Industrial Enterprise Code
`email`	string	Not set	E-mail
`bank_id`	number	Not set	Bank ID
`bank_account`	string	Not set	Bank account ID
`comments`	string	Not set	Comment
`personal_manager`	string	Not set	Personal manager
`connect_date`	number	Not set	Connection date in the Unix timestamp format

Element	Type	Default value	Description
`is_send_invoice`	0, 1	0	**Send invoice over email** parameter (1 for yes, 0 for no)
`advance_payment`	0, 1	0	**Advance payment** parameter (1 for yes, 0 for no)
`switch_id`	number	Not set	Switch ID
`port_number`	number	Not set	Switch port number
`binded_currency_id`	number	Not set	ID of the user's preferred currency
`parameters`	collection of `parameter` elements (see **parameter** on page **217**)		
`groups`	collection of `group` elements (see **group** on page **217**)		

account

The `account` element may contain:

Element	Type	Default value	Description
`id`	number	Not set	Account ID (reserved for future use). Not related to the actual ID assigned to the account once recorded into the database.
`is_blocked`	number	0	see **Block type**
`balance`	real number	0	Account balance
`credit`	real number	0	Credit
`vat_rate`	real number	0	VAT rate
`sale_tax_rate`	real number	0	Sale tax rate
`int_status`	0, 1	1	Internet status (0 if off, 1 if on).

Block type

- `0` – not blocked;
- `256` – system block;
- `768` – system block, adjust recurring fee;
- `1280` – system block, adjust prepaid traffic;
- `1792` – system block, adjust recurring fee and prepaid traffic;

parameter

The `parameter` element must contain:

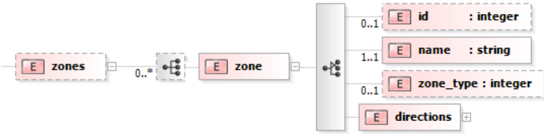
Element	Type	Default value	Description
`parameter_id`	number	Mandatory element	ID of an additional parameter
`parameter_value`	string	Mandatory element	Parameter value

group

The `group` element must contain:

Element	Type	Default value	Description
`group_id`	number	Mandatory element	ID of the group to which the user belongs

Elements: zone

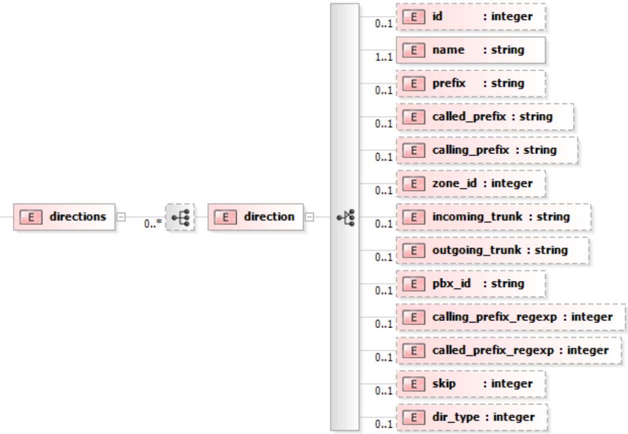


Contains description of a telephone zone. May include:

Element	Type	Default value	Description
`id`	number	Not set	Zone ID (reserved for future use). Not related to the actual ID assigned to the zone once recorded into the database
`name`	string	Mandatory element	Zone name
zone_type	number	0	Calls type: 0 – local, 1 – inner-zone, 2 – inter-city, 3 – international
`directions`	collection of `direction` elements (a)	Mandatory, though may be empty	Each `direction` element must contain `id` element whose value is a number; these are the IDs of the included directions

a. Not to be confused with the second-level element `directions` and its child `direction`.

Elements: direction


E directions
0..*
E direction
0..1 E id : integer
1..1 E name : string
0..1 E prefix : string
0..1 E called_prefix : string
0..1 E calling_prefix : string
0..1 E zone_id : integer
0..1 E incoming_trunk : string
0..1 E outgoing_trunk : string
0..1 E pbx_id : string
0..1 E calling_prefix_regexp : integer
0..1 E called_prefix_regexp : integer
0..1 E skip : integer
0..1 E dir_type : integer

Contains description of a telephone direction. May include:

Element	Type	Default value	Description
`id`	number	Not set	Direction ID. Used only for cross-references from the `zone` elements within the same file. Not related to the actual ID assigned to the direction once recorded into the database
`name`	string	Mandatory element	Direction name
`prefix`	string	Mandatory element	Regular expression to include matching numbers in the direction
`prefix`	string	Not set	Same as called_prefix (left for backwards compatibility; if both are set, called_prefix overrides `prefix`)
called_prefix	string	Not set	Prefix or regular expression for checking the called number
calling_prefix	string	Not set	Prefix or regular expression for checking the calling number
zone_id	number	Not set	ID of the parent zone
incoming_trunk	string	Not set	Incoming trunk name
outgoing_trunk	string	Not set	Outgoing trunk name
pbx_id	string	Not set	PBX ID
calling_prefix_regexp	number	1	How to interpret calling_prefix: 0 – prefix, 1 – regexp

Element	Type	Default value	Description
called_prefix_regexp	number	1	How to interpret called_prefix: 0 – prefix, 1 – regexp
skip	number	0	1 – skip this direction (identify no calls into it)
dir_type	number	Same as in the containing zone, or 0 if `zone_id` is not set	Calls type: 0 – local, 1 – inner-zone, 2 – inter-city, 3 – international

A direction must be defined with at least one of the following elements: called_prefix, calling_prefix, incoming_trunk, outgoing_trunk, or pbx_id.

Example XML file

Import of the example file given below creates the following entities in UTM:

- Telephone direction “Texas” defined by a regexp that checks that the called numbers are 11-digit and start with 1 followed by `713`, `432`, or `281`.
- Telephone zone “USA” of type 3 (international) including the telephone direction “Texas”.
- User F.A. Cotton, an individual belonging to the group 2, with certain login and password, having one account with balance of $ 20.5.

```
<?xml version="1.0" encoding="utf-8"?>
<import>
  <users>
    <user>
      <accounts>
        <account>
          <id>1</id>
          <balance>20.5</balance>
          <vat_rate>0.10</vat_rate>
          <int_status>1</int_status>
        </account>
      </accounts>
      <id>11</id>
      <login>cotton</login>
      <password>aipsw123</password>
      <full_name>F.A. Cotton</full_name>
      <is_juridical>0</is_juridical>
      <groups>
        <group>
          <group_id>2</group_id>
        </group>
      </groups>
    </user>
  </users>
  <zones>
    <zone>
      <id>1</id>
      <name>USA</name>
      <zone_type>3</zone_type>
      <directions>
        <direction>
          <id>1</id>
        </direction>
      </directions>
    </zone>
  </zones>
  <directions>
    <direction>
      <id>1</id>
      <name>Texas</name>
      <called_prefix>^1(713|432|281)[0-9]{7}$</called_prefix>
      <called_prefix_regexp>1</called_prefix_regexp>
    </direction>
  </directions>
</import>
```

16

DEALER MODULE

Introduction

Dealer module provides the interface for creating and operating dealers. The module consists of the following parts:

- core library `liburfa-dealer.so` containing the dealer functionality;
- part of the administrator's interface responsible for handling dealers;
- dealer interface proper.

Dealer is a system object providing an ability to connect to the billing system and perform some administrative operations regarding a particular subset of users. Dealer's interface is a Java application based on UTM Control Center and analogous to the administrator's interface, but with limited functionality.

Creation and operation of dealers require a separate license. To verify the availability of the license and its term of validity, see About: Licenses in the UTM5 administrator's interface and check the list item denoted **Dealer interface**.

The creation of dealers and their abilities are described below.

Creating dealers

Creation of dealers is performed in the administrator's interface under Users and groups: Dealers. Technically, a dealer is treated by the system as if it were a system user hardwired to the special system group **Dealers**. The properties of this system group (in particular, the list of permitted operations), as well as those of other system groups, may be checked under Users and groups: System groups.

*The **Dealers** system group is built-in, so its properties can not be changed via the administrator's interface.*

Among other dealer's properties there is an **Access rights** group of parameters that regulates the access of this particular dealer to the following entities:

- Users (see **Users** on page **26**);
- Accounting periods (see **Accounting periods** on page **29**);
- Services (see **Services** on page **31**);
- Tariffs (see **Tariff plans** on page **30**);

- Houses (the list of registered buildings).

For each of these kinds, a list of options is provided to set up dealer's access to each entity individually. By default the access to all of them is denied, except for the users (if any) created by this very dealer.

The other available group of pages is the **Reports** group. There are reports for users who are assigned to the selected dealer. The following reports are available in this group:

- General report;
- Report on blockings.
- Traffic report;
- Telephony report;
- Sessions report;
- Report on payments;
- Report on services;
- Report on invoices.

Each user may be attached to only one dealer.

Dealer's access to a particular user may also be set up on the user's properties page using the **Link to dealer** button (see **Users and groups: Users** on page **44**).

The creation of dealers and setting up their privileges are described in the following examples: **Creating dealers** on page **140**, **Setting dealer's permissions** on page **141**, and **Linking users to dealers** on page **141**.

Once created and set up, the dealer may act as an administrator with limited rights in respect of the certain subsets of users and other entities. In particular, a dealer may perform the following operations (see the corresponding examples in **Usage examples** on page **129**):

- Create and delete users;
- Change the user's properties, except for the remote switch, preferred currency, technical parameters, group attribution, and other dealers' access to this user;
- Create, change, and remove accounts;
- Create, change, and remove service links;
- Create, change, and remove tariff links;
- Link users to houses;
- Make payments;
- Compose reports;
- Change own password.

Dealer interface

Installation and startup

1. Download the dealer's interface from the client's personal cabinet on **https://www.utm-billing.com/customer.php** under **Downloads**. The file is called `utm_dealer.zip`.
2. Unpack the archive on the dealer's workstation.

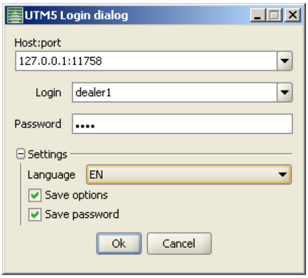
To use the dealer's interface, Java Runtime Environment (JRE) version 8.0 (Java 1.8.x) or above is required.
JRE distributive is available for free at **http://j-ava.com**.

3. Start the control center either by clicking on the file `utm_dealer.jar` or from the command line by executing

```
java -jar utm_dealer.jar
```

The login dialog window similar to that in the administrator's interface will appear.

4. Enter the IP address and colon-separated port number to connect to. If the port number is omitted, the default value of 11758 is assumed.



Host:port 127.0.0.1:11758
Login dealer1
Password ••••
Settings
Language EN
☑ Save options
☑ Save password

5. Enter the login and password of the dealer specified at the time of its creation in the administrator interface.
6. In the **Settings** group of parameters select the language to use.

Note that the selected language is not applied immediately to the login dialog itself. Instead, the language switch occurs on the next launch of the program.

7. Check **Save options** if you want to save the parameters just entered (except for password) in the settings file for use during subsequent launches. Check **Save password** if you also want to save the password as well.

It is highly recommended to change the dealer's password immediately after logging in for the first time (see ***Additional*** *on page* ***225****).*

Items of the dealer interface are listed below by chapters, as they appear on the left pane.

Users

This page contains a list of users accessible for the dealer, containing the following info about each user:

- **User ID –** is the ID of the user in the system.
- **Login –** is the user's login.
- **Primary account –** is the account number.

- **Full name –** is the full name of the user or a title of the legal entity.
- **Block ID –** is the blocking status of the user.
- **Balance –** is the account balance.
- **IP (VPN) and IP (non-VPN) –** are the user's IP addresses.

An interface for adding, editing, or removal of users, as well as making payments, is included.

Dealer has neither editing nor viewing access to the lists of dealers, system users, groups, and system groups.

The Add and Edit buttons open the user details window similar to that of the administrator's interface (see **Administrator's interface: Users** on page **44**). The window includes a number of interface pages accessible via the quick links on the left pane, which are gathered into the following groups:

User

- **Main –** contains login, full name, password, and the following interface elements:
  - **Payment in advance** check box;
  - **Generate document for user** button that displays the handout document for the user containing login, password, and the provider's contacts;
- **Additional –** contains bank account data, etc.
- **Contacts –** contains personal data (address, phone, e-mail) of the contact person.
- **Additional contacts –** contains personal data of additional contact persons, if any.
- **Additional info –** contains view-only auxiliary information (dates of creation and last modification of the user).

Dealer has neither editing nor viewing access to the user's group membership, contracts, preferred currency, and technical parameters.

Tariffication

- **Accounts –** contains the list of the user's accounts.
- **Service links –** contains the list of user's service links.
- **Tariff links –** contains the list of the user's tariff plans.
- **Technical parameters –** are the arbitrary parameters associated with the user. Their values may be used in the commands for controlling the external software, which are sent by UTM5 as a response to certain events, see **UTM5 RFW: Firewall rules** on page **186**.

While dealing with the user's accounts, service links, and tariff links, the dealer may use only the explicitly stated subset of services, tariffs, and accounting periods.

Reports

This group of pages contains interface for creating reports. The possible reports are the same as those listed in the **Reports** top-level group (see below), with the only distinction that they are based on the data related to the currently selected user, rather than all users.

The Delete button next to the list of users removes the user, once the related service links and tariff links are removed, or displays an error message otherwise.

The Search button opens the search window (see **Search page** on page **107**).

The New payment button opens the payment window (see **Payment page** on page **106**).

Reports

Dealer interface supports a variety of reports essentially similar to those presented in the administrator interface (see **Administrator's interface: Reports** on page **47**), including:

- General report;
- Traffic report;
- Report on services;
- Telephony report;
- Sessions report;
- Report on payments;
- Report on blockings.

Reports for all users are limited to those users which the dealer has access to.

Unlike administrator, the dealer can not compose reports limited to custom groups of users.

Additional

This page contains interface for changing the dealer's password. The **Change** button turns active only if **New password** and **Confirm new password** coincide.

About

This page displays the program version number and the general info related to the dealer.

CASHIER MODULE

Introduction

Cashier is a system user with the primary capability of making payments. Cashier's interface is a Java application based on UTM Control Center and analogous to the administrator's and dealer's interfaces, yet with even more limited functionality.

Creation and operation of cashiers require a separate license. To verify the availability of the license and its term of validity, see About: Licenses in the UTM5 administrator's interface and check the list item denoted **Cashier interface**. The license may or may not set a limit for the number of cashiers simultaneously connected to the UTM5 core.

There is also an alternative cashier interface (see **Alternative interface** on page **230**) which is maintained for the sake of backward compatibility and licensed separately.

The creation of cashiers and their abilities are described below.

Creating cashiers

Like other system users, cashiers are created via the administrator's interface under Users and groups: System users. A system group with appropriate permissions should be set up (see **System groups** on page **50**) for the cashiers. The necessary functions (see the list below) are grouped in a separate branch on the tree view.

FID	Function name	Description
`0x1206`	`rpcf_search_users_new`	Searches for users
0x2006	rpcf_get_userinfo	Returns information about a user
0x2026	rpcf_get_user_by_account	Returns user ID for a personal account ID
`0x2033`	`rpcf_get_user_account_list`	Returns the list of user IDs
0x212c	rpcf_get_cashier_settings	Returns cashier's interface settings
0x2600	rpcf_get_accounting_periods	Returns accounting periods list
`0x2910`	`rpcf_get_currency_list`	Returns the list of currencies
`0x3008`	`rpcf_payments_report_owner_ex`	Generates the report on payments made by the current

FID	Function name	Description
`0x3100`	`rpcf_get_payment_methods_list`	Returns the list of payment methods
`0x3110`	`rpcf_add_payment_for_account_notify`	Makes a payment and emails the customer about it
`0x440A`	`__rpcf_whoami`	Returns the current system user's info
0x11112	rpcf_get_core_time	Returns system time
0x15109	rpcf_get_accountinfo	Returns personal account information

The subnet mask to login from may also be specified.

Besides the group-defined permissions, the s' abilities depend on the interface settings (see **Cashier interface** on page **99**).

Once created and set up, a can:

- Make payments;
- Compose reports on payments.

Cashier interface

Installation and startup

1. Download the cashier's interface from the client's personal cabinet on **https://www.utm-billing.com/customer.php** under **Downloads**. The file is called `utm_.zip`.
2. Unpack the archive and run the `utm_.jar` file in a manner similar to starting the administrator's interface (see **Installation and startup** on page **129**).

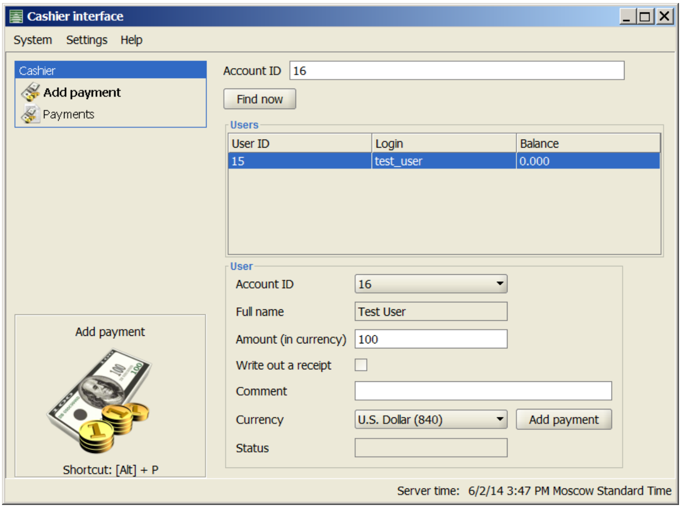
Java Runtime Environment (JRE) version 8.0 (Java 1.8.x) or above is required in order to use the cashier's interface.
JRE distributive is available for free at **http://j-ava.com**.

Interface elements and their behavior are compatible with those described in **Administrator's interface: Common features** on page **43**.

Items of the interface are listed below by chapters, as they appear on the left pane.

Payment

The **Add payment** page contains the interface for searching the users and making payments.



Account ID: 16

Users

User ID	Login	Balance
15	test_user	0.000

User

Account ID: 16

Full name: Test User

Amount (in currency): 100

Write out a receipt: ☐

Comment:

Currency: U.S. Dollar (840)

Status:

Server time: 6/2/14 3:47 PM Moscow Standard Time

Select a user in the search results to make a payment. The cashier can not access the complete list of registered users.

The search may be done by the user ID, account ID, login fragment, or name fragment (unless some of these options are forbidden by the administrator). The number of users to display in search results may also be limited by the administrator's interface settings.

Payment method is set to **Cash**. Things left for the cashier to select are: account number, if the client got more than one, currency (from the administrator-defined list) and the amount. Check Write out a receipt if you want to print a receipt for the operation. One may add a comment to the payment in the **Comment** field.

Reports

Cashier's interface supports the report on payments similar to that presented in the admin interface (see **Administrator's interface: Report on payments** on page **78**), except for:

- Only the payments made by this cashier are included;
- Selection by users group is disabled;
- The "Payment method" and "Received by" columns are removed as irrelevant;
- Context menu items "Print receipt" and "Roll back" are not available.

Alternative interface

The alternative cashier’s interface exists in basic and advanced versions. The minimum set of necessary functions is given below:

FID	Function name	Description
`0x1202`	`rpcf_search_users_lite`	Searches for users by login substring (returns five first results)
`0x2001`	`rpcf_get_users_list`	Returns the list of user attributes a. Required only for the extended interface.
`0x2011`	`rpcf_get_users_count`	Returns the number of users (a)
`0x2030`	`rpcf_get_accountinfo`	Returns the user’s account info
`0x2033`	`rpcf_get_user_account_list`	Returns the list of user IDs
`0x2910`	`rpcf_get_currency_list`	Returns the list of currencies
`0x3008`	`rpcf_payments_report_owner`	Generates the report on payments made by the current
`0x3100`	`rpcf_get_payment_methods_list`	Returns the list of payment methods
`0x3110`	`rpcf_add_payment_for_account`	Makes a payment
`0x440A`	`__rpcf_whoami`	Returns the current system user’s info

Installation and startup

1. Download the cashier’s interface from the client’s personal cabinet on **https://www.utm-billing.com/customer.php** under cashier **’s interface**. The file is called either `cashier.zip` (basic version) or `cashier_ext.zip` (extended version).
2. Unpack the archive on the cashier’s workstation.

To use the cashier’s interface, Java Runtime Environment (JRE) version 5.0 (Java 1.5.x) or above is required.
JRE distributive is available for free at **http://j-ava.com**.

3. On the same page of the client’s personal cabinet download the `ccse.keystore` file and place it in the same folder where the cashier’s interface has been just unpacked.
4. On the same page of the client’s personal cabinet download also the private key file `privkey.pem` and the certificate `cert.crt`, and place both to the server where the UTM5 core is running.
5. Add the following lines to `utm5.cfg`:

```
ssl_cert_file=<path to cert.crt>
ssl_privkey_file=<path to privkey.pem>
ssl_privkey_passphrase=<password set in the personal cabinet>
```

6. Restart the UTM5 core.

7. Start the control center either by clicking on the file `control.center.se.jar` or from the command line by executing

```
java -jar control.center.se.jar
```

The login dialog window will appear.

Language of the control center is set according to the current system locale.

8. Enter the IP address and colon-separated port number to connect to. If the port number is omitted, the default value of 11758 is assumed.
9. Enter the login and password of the selected cashier as set in the administrator's interface.

The cashier's interface includes two pages accessible via quick links on the left pane, namely, **Customers** and **Payments**.

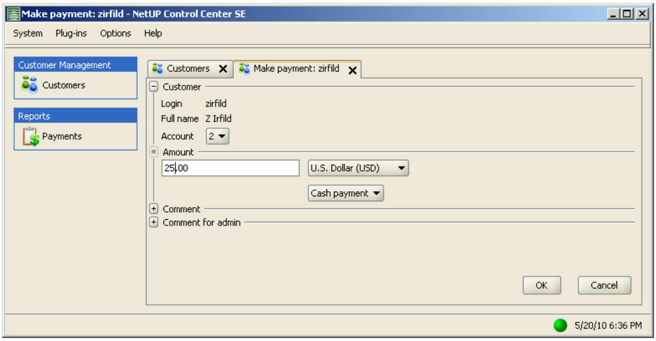
Interface elements and their behavior are generally compatible with those described in **Administrator's interface: Common features** on page **43**, except for the following features:

- Context menu of the lists does not provide the **Export** option;
- Status line in the bottom part of the window contains not only time and date, but also the connection status.

Customers

This page contains the search interface. In the basic version, the search is restricted to at most five results, done only by login, and returns an empty set to an empty search request. In the extended version, the search may be done by any field, is unrestricted in terms of result entries, and returns all users when searching for an empty string.

The payment is made on a separate page called **Make payment**, which opens by double-clicking on an entry in the search results list, or via the context menu item of the same name.



Login zirfild
Full name Z Irfild
Account 2
25.00
U.S. Dollar (USD)
Cash payment
5/20/10 6:36 PM

The **Make payment** page contains the following parameters:

- **Login –** of the user (view-only).
- **Account –** to be selected from a drop-down list if the given user has multiple accounts.
- **Amount –** of payment.
- **Currency**
- **Method –** of payment.
- Arbitrary **Comment** s (**for admin** and for the user).

Payments

This page contains the interface for creating a report on payments similar to that described in **Administrator’s interface: Report on payments** on page **78**, except for:

- Only the payments made by this cashier are included;
- Selection by users group is disabled;
- Filter option is disabled;
- The “Payment method” and “Received by” columns are removed as irrelevant;
- Context menu items “Print receipt” and “Roll back” are not available.

18

UTM5 TRAY UTILITY

Introduction

For a more convenient access to the personal account balance use the `utm5_tray` utility. This program runs on the client PC and refreshes the balance information and data on remaining prepaid traffic periodically. Also, with this tool the Internet access may be switched on and off.

The utm5_tray utility is an alternative for the web interface (see **Web interface** on page **239**) which mostly covers the same functionality.

Installation and startup

To install and start the `utm5_tray` utility:

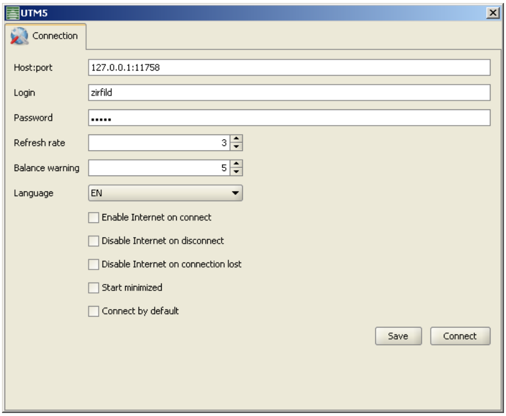
1. Download the user's interface from the client's personal cabinet on **https://www.utm-billing.com/customer.php** under **Downloads**. The file is called `utm5_tray.zip`.
2. Unpack the archive on the user's computer.

Java Runtime Environment (JRE) version 6.0 (Java 1.6.x) or above is required to launch the utility.
JRE distributive is available for free at **http://j-ava.com**.

3. Start the utility either by clicking on the file `utm5-_tray.jar` or from the command line by executing

```
java -jar utm5-_tray.jar
```

The login dialog window will appear.



Host:port 127.0.0.1:11758
Login zirfild
Password •••••
Refresh rate 3
Balance warning 5
Language EN
☐ Enable Internet on connect
☐ Disable Internet on disconnect
☐ Disable Internet on connection lost
☐ Start minimized
☐ Connect by default

4. Enter the IP address and colon-separated port number to connect to. If the port number is omitted, the default value of 11758 is assumed.
5. Enter the login and password of the user specified at the time of its creation in the administrator interface.
6. In the **Refresh rate** field enter the timeout for refreshing the status information (in seconds), or leave the default value.

7. In the **Balance warning** field enter the balance value (in the user’s preferred currency) to issue a warning when reached, or leave the default value.
8. In the **Settings** group of parameters select the language to use.

Note that the selected language is not applied immediately to the login dialog itself. Instead, the language switch occurs on the next launch of the program.

9. Set the flags **Enable Internet on connect** and/or **Disable Internet on disconnect** if you wish to tie the connection status to utm5_tray being started.
10. Set the flag **Disable Internet on connection loss** if you want the billing system core to disable Internet for you upon loss of connection.
11. Set the flag **Start minimized** if you wish to minimize the utm5_tray window right after starting up.
12. Set the flag **Connect by default** if you wish to start utm5_tray on computer startup.
13. Press **Connect**. The main utm5_tray window will show up.

Once started, the program is represented by an icon in the system tray (see **Tray icon** on page **236**). When the main window is closed, the program still persists in memory and may be activated by double-clicking on this icon.

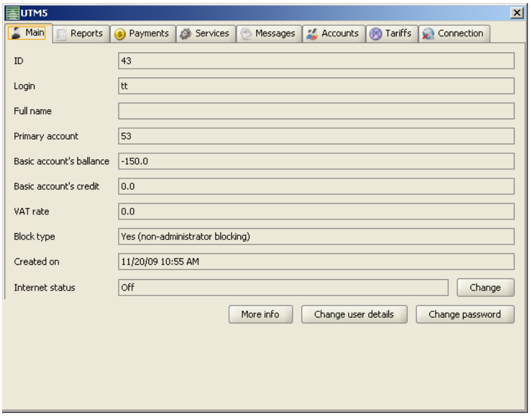
Interface pages

The main window of utm5_tray contains several tabs with quick links at the top. The user’s access to particular tabs is controlled by the administrator (see **Administrator’s interface: Tray settings** on page **98**).

Main

The **Main** tab displays the following information:

- User ID;
- Login;
- Full name;
- Primary account number;
- Primary account balance;
- Primary account credit;
- VAT rate;
- Blocking type;
- Creation date;
- Internet status (with the ability to change it, unless the account is blocked for running out of money).



ID	43
Login	tt
Full name	
Primary account	53
Basic account's ballance	-150.0
Basic account's credit	0.0
VAT rate	0.0
Block type	Yes (non-administrator blocking)
Created on	11/20/09 10:55 AM
Internet status	Off

Also, the page contains the following buttons:

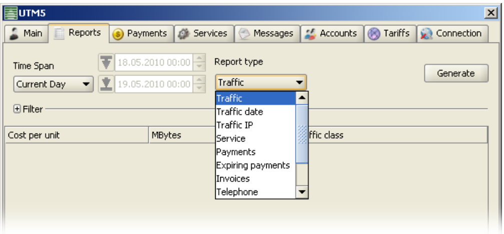
- **More info –** opens the window with additional user info (passport data, address, contacts);
- **Change user details –** opens the window for editing user info. Name, address, passport data and the bank details are write-only, that is, may be entered only once and can not be edited afterwards. The contact information (which may include phone number, e-mail, web address, ICQ number) may be edited at any time.
- **Change password –** provides the interface for changing the user's password.

Reports

The **Reports** tab contains interface for composing reports on given user, largely similar to that in the administrator's interface (see **Reports** on page **47**).

The following types of reports are included:

- Traffic;
- Traffic grouped by date;
- Traffic grouped by IP;
- Services;
- Payments;
- Expiring payments;
- Invoices;
- Telephony;
- VPN;
- Blockings;
- Internal transfer.



Time Span: Current Day

18.05.2010 00:00

19.05.2010 00:00

Report type: Traffic

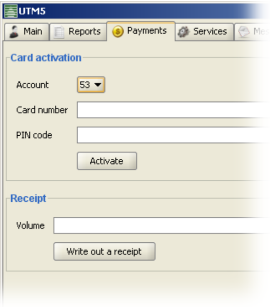
The user's access to particular kinds of reports is controlled by the administrator (see **Administrator's interface: Tray settings** on page **98**).

Payments

The **Payments** tab contains the following interface elements:

Under **Card activation** the user can make a payment by entering the number and PIN code of the prepaid card.

Under **Receipt** the user may compose a receipt with arbitrary sum for printing.



Account: 53

Services

This tab contains the list of services attached to the user’s accounts. For each service the list contains its name, type, the tariff plan where it belongs (if any), accounting period, and price per period.

Messages

This tab contains interface for exchanging system messages. Its counterpart on the administrator’s side is described in **Administrator’s interface: Messages** on page **53**.

Accounts

This tab contains the balance of all user accounts. There are also buttons for changing the internet status and for imposing voluntary blocking, if this is permitted by the administrator’s settings and the user balance is above the required threshold.

Tariffs

This tab contains the list of user’s tariff plans. There are also buttons for viewing the tariff plan history and for switching to another tariff plan, if this is permitted by the administrator’s settings and the user balance is above the required threshold.

Connection

This tab contains the same interface elements as the login dialog window. Any changes are applied at the next launch of the program.

Tray icon

When utm5_tray is working, an icon appears in the system tray. Depending on the connection to the system core and the Internet status, the icon may look as follows:

- ▤ – connection is being established (during the launch of the program).
- 🟩 – connection established, Internet on.
- 🟨 – connection established, some of the user’s accounts are blocked.
- 🟥 – connection established, Internet off.

On mouse hover the icon displays a pop-up message containing the balance of the user’s primary account. Also, a warning message pops up when the balance reaches zero.

The icon has a context menu of its own, which contains the following items:

- **Open –** activates the main window of the program.
- **Enable / Disable Internet –** turns Internet on and off.
- **About –** displays the program version number and contact info.
- **Exit –** closes the program.

19

WEB INTERFACE

Introduction

UTM5 provides web interface, where an end user may get a statement of account balance, make a payment via a prepaid card, and perform some other activities.

There is an alternative client covering the same functionality, which comes in a form of a standalone program (see **UTM5 tray utility** on page **233**).

Installation

UTM5 web interface is installed as described below.

Running the web interface requires PHP 5.4 or newer version.

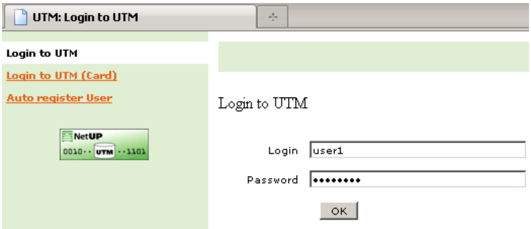
1. Download the server portion of web interface (file called `utm5_web_php.zip`) from the client’s personal cabinet on **https://www.utm-bi-lling.com/customer.php** under **Downloads**.
2. Unpack the archive on the web server.
3. In the web interface config file (see **Config files: Main** on page **243**) enter the address, port, login, and password for access to the UTM5 core, as well as the path to where the archive has been unpacked.
4. In the PHP config file (commonly `php.ini`) set the following parameters:
    - `short_open_tag` must be set to `1` (possible equivalents are: `On`, `True`, and `Yes`);
    - time zone must be set correctly, for example:

      ```
      date.timezone ='America/Chicago'
      ```

The web server must be capable of executing PHP files. Depending on the OS and on the PHP interpreter version, installation of additional OpenSSL and PCRE extensions for PHP may be required.

Entrance page

To enter the web interface, launch an Internet browser, open the root page (by default, it is located at **h-ttp://your.server/utm5_web_php**, where `your.server` stands for the domain name of the UTM5 server) and enter the user’s login and password into the corresponding fields.



Login to UTM

Login: user1

Password: ••••••••

To enter using a prepaid card, enter the card number and PIN for login and password. A card user will be created (see **Card users** on page **48**) with login `card_NUM`, where `NUM` is replaced with the card number. User’s password is set to the card PIN.

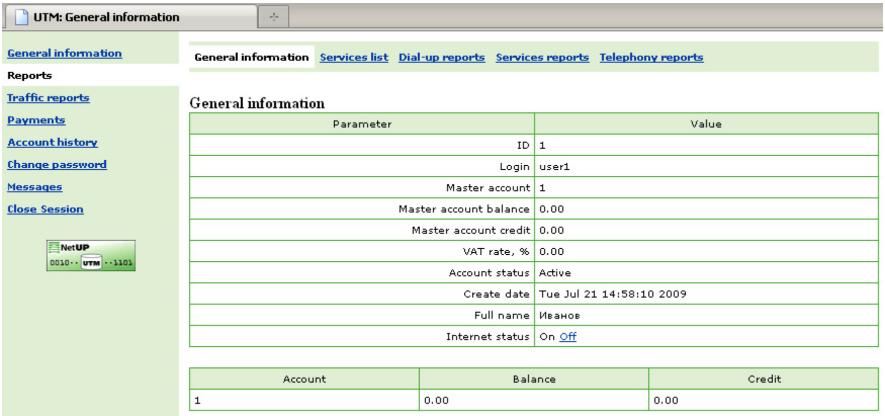
To enter using hotspot, switch to the page **Entrance (hotspot)** (unless you’ve been already redirected there automatically) and enter the card number and PIN for login and password. The session details page will show up to display the remaining time, issued IP address and other session information. Simultaneously, the page requested before will open in a separate window.

Interface pages

Common

User

First tab contains the general information, including user’s ID, login, full name, creation date, summary balance of all accounts, and state.



General information

Parameter	Value
ID	1
Login	user1
Master account	1
Master account balance	0.00
Master account credit	0.00
VAT rate, %	0.00
Account status	Active
Create date	Tue Jul 21 14:58:10 2009
Full name	Иванов
Internet status	On Off

Account	Balance	Credit
1	0.00	0.00

Additional tab contains more info on the user, including address, contact information, and bank details.

Edit tab contains the interface for editing some of the user properties. Name, address, passport data and the bank details are write-only, that is, has to be entered only once and can not be edited afterwards. The contact information (which may include phone number, e-mail, web address, ICQ number) may be edited at any time.

Accounts

This page contains info on all the user’s accounts. For each account, it displays its ID, balance, credit, tax rates, blocking state and Internet state. Blocked amount is the sum of charges for the services not used because of the account blocking, which is due to be returned to the user by the end of the period. Besides that, a number of actions can be applied to any account:

- **Return –** immediately the blocked amount, if any;
- **Turn Internet on –** (if it is off and the account is not blocked);
- **Turn Internet off –** (if it is on);
- **Make promised payment –** which is a credit payment with limited sum and fixed due date. A commission fee may be charged for using this facility. The credit parameters (maximum sum, due date, minimum interval between promised payments, minimum balance,

commission fee, minimum balance to use the facility without fee) are set up by the administrator (see **Interfaces: Promised payments** on page **101**).

- **Voluntary suspension –** which is a temporal suspension that a user may impose on oneself, for example, to save the periodic fee for the time span when the service is not required. A commission fee may be charged for using this facility. The suspension parameters (minimum and maximum time, minimum interval between suspensions, ability to unblock oneself early, minimum balance, commission fee, minimum balance to use the facility without fee) are set up by the administrator (see **Interfaces: Voluntary suspension** on page **101**). Note that after using a voluntary suspension on oneself the user has to turn Internet on manually (this is done on the same page, see **Turn Internet on** above).
- **Internal transfer –** of funds between one's accounts.

Password

This page contains the interface for changing the user's password to all password-protected services, as well as to the web interface itself.

Messages

This page contains interface for exchanging system messages. Its counterpart on the administrator's side is described in **Administrator's interface: Messages** on page **53**. The messages are divided into Incoming, Outgoing, and New, and also can be filtered by time.

Reports

This page contains the following user's reports, each on a separate tab:

- Traffic report;
- Services report;
- Payments report;
- Invoices;
- Telephony report;
- Hotspot report;
- Blockings report;
- Internal transfer;
- Other charges.

Traffic classes removed by the administrator would still show up as existing in the web interface traffic report.

The functionality of reports is similar to that of the administrator's interface, except for the context menu.

Tariffs and services

Tariffs tab contains the list of user's accounts together with the tariff plans linked to each. For each tariff plan, it displays the start and end dates. The user may choose to switch to another compatible tariff plan (see **Tariff plans compatibility** on page **30**) starting from the next accounting period. A commission fee may be charged for using this facility (see **Interfaces: Tariff switch** on page **100**).

Services tab contains the list of all services attached to the user's accounts. For each service, it contains the start and end dates, price, and the amount charged for the current period.

Tariff plan history tab contains the list of all tariff plans attached to the user's accounts in the past.

Payments

Card activation tab page is where the user can make a payment by entering the number and PIN code of a prepaid card.

Invoices tab contains the interface of generating a receipt for printing.

The rest of tabs (if any) contains the interface of making payments via various payment systems, one per tab. This functionality is switched on/off and otherwise controlled by the separate config file, see **Config files: Payment systems** on page **244**.

Exit

This hyperlink performs an immediate logout and returns to the authorization page.

Config files

Main

By default, UTM5 web interface uses the config file located at `/lib/config.php` relative to the web interface root folder. Below is the list of possible parameters.

Parameter	Possible values	Default value	Description
`$CONF_DEFAULT_MODULE1`	string	`00_user`	Default page after user login
`$CONF_DEFAULT_MODULE2`	string	`card`	Default page after card login
`$CONF_DEFAULT_MODULE3`	string	`hotspot`	Default page after hotspot log-in
`$CONF_DEFAULT_LOGIN`	string	`00_login`	Default page before login
`$CONF_PATH`	string	`/utm5/`	Relative path from site root to the web interface

Parameter	Possible values	Default value	Description
`$CONF_WEB_USER`	string	`web`	System user login
`$CONF_WEB_PASS`	string	`web`	System user password
`$CONF_CORE_HOST`	IP address	`127.0.0.1`	Address of the UTM5 core host
`$CONF_CORE_PORT`	number	`11758`	Port number to connect to the core
`$CONF_LANG`	ru, en	`ru`	Web interface language
`$CONF_REDIRECT_HOTSPOT`	0, 1	`0`	Enables redirect after the hotspot login to the previously requested page

Additional modules

Several modules located at `/modules` (in particular, the module of promised payments called `promised_payment.php` and the internal transfer module `funds_flow.php`) contain some config parameters of their own. The parameter named `$MOD_VISIBLE` must be set to `true` in order to include the corresponding module in the web interface.

Payment systems

A separate config file named `/lib/payment_systems_config.php` controls payment systems. It contains the following parameters (among others):

Parameter	Possible values	Default value	Description
`$web_money_visible`	true, false	false	Switches on WebMoney support
`$yandex_money_visible`	true, false	false	Switches on Yandex Money support
`$chronopay_visible`	true, false	false	Switches on Chronopay support
`$mobi_money_visible`	true, false	false	Switches on MobiMoney support
`$web_creds_visible`	true, false	false	Switches on WebCreds support

Each of the above-mentioned lines is followed by several parameters specific to the corresponding payment system.

HOTSPOT MODULE

Common features

Hotspot module is intended to provide Internet access paid by the hour. Operation of the hotspot module requires a separate license. To verify the availability of the license and its terms of validity, see About: Licenses in the UTM5 administrator's interface and check the list item denoted **Hotspot module**.

The hotspot module may work either with the UTM5 web interface (see **Web interface** on page **239**), or via RADIUS authorization.

Once the user is authorized, the web interface page starts refreshing automatically at regular intervals. If the refresh does not happen in due time because the user has closed the authorization page, the session expires. The expiration may also occur due to running out of money. When the session is either expired or explicitly closed by the user selecting **Close** in the menu, the Internet access is blocked and the user is charged for the session's duration. The session lifetime is set by the `hotspot_refresh_timeout` parameter (see **Core settings: Interface parameters** on page **152**).

To use the hotspot module, one has to create a tariff plan containing the hotspot service (see **Hotspot service** on page **65**). Price per hour may be time-dependent. A limited list of allowed networks to login from, as well as the maximum connections number for the given login, may also be specified.

To use the hotspot module along with prepaid cards, it is necessary to create a pool of cards and connect them to the tariff plan containing the hotspot service (see **Generating cards: Tariff ID** on page **51**). When issued a card, a user should in the first place register it in "Auto register user" section of the web interface and thus obtain a login and password, which are subsequently used for authorization in "Login to UTM (Card)" section (see **Web interface: Entrance page** on page **240**).

If the hotspot access has to be charged per traffic rather than per hour, the hotspot service must be linked to an IP traffic service by checking the Dynamic IP address allocation option for both of them. At that, user authorization on the UTM5 web interface would require the login and password stored in the properties of the hotspot service link.

UTM5 DHCP

Introduction

NetUP UTM5 DHCP functions as a DHCP server. It receives messages and processes them according to RFC 2132. It uses the following entities: switch type, switch, DHCP pools and IP Groups.

UTM5 DHCP allows one to associate a static IP address or a dynamic address pool with a MAC address, switch or a certain switch port.

UTM5 DCHP only allows to assign IPv4 addresses.

UTM5 DCHP uses data from the database, communicates with the UTM5 Core via the Stream protocol and is able to receive messages about changes in the database and the need to update certain information.

UTM5 DCHP always works in not authoritative mode. If the server is unable to determine the client's configuration, based on DHCP query parameters, it ignores the query.

UTM5 DCHP module allows a maximum of 10 simultaneous DHCP leases without a module license. You have to buy a separate license for this module, if you want to serve more clients.

Entities, used by UTM5 DHCP

1. **Switch type** - an entity that contains a certain switch type parameters. Those parameters are:
   - **Name –** a string containing the name of the switch. Uniqueness is not mandatory, but is recommended
   - **Supported volumes –** the number of ports for this switch type. May be several numbers, separated by commas
   - **DHCP option 82 parameters –** description of DHCP option 82 parameters, used by this switch type for composing DHCP requests:
     - **Remote ID –** is the ID of the switch, acting as a DHCP relay, which the request came from
     - **Port –** is the port number of the switch, acting as a DHCP relay, which the request came from
     - **VLAN ID –** is the VLAN ID, if there is one

These parameters have properties like parameter type (string/binary), disposition, offset and length. These properties are used to read those parameters from the option 82 of a DHCP request.

1. **Switch** - an entity that contains a certain switch parameters. Those parameters are:
   - **Name –** is the name of the switch. The name should help identifying the switch and its location, and at least it should be unique (this is not mandatory, but is recommended)
   - **Actual address –** is the actual address where one can find this switch
   - **Type –** is the internal Switch type ID, which contains the parameters for this switch type
   - **Remote ID –** is the Remote ID parameter of the DHCP option 82. It is used for composing a DHCP request. The parameter type and its length are set in the corresponding Switch type
   - **Ports count –** is the number of ports for this particular switch
   - **Access parameters –** are IP, login and password for the switch

These parameters may be used in the firewall rules (UTM5 RFW module), which have to do with sending commands to the router. E.g. when one needs to turn off a switch port to prevent the customer from using Internet, when the customer has run into debt.

These parameters are associated with the following variables:

- USW_IP
- USW_LOGIN
- USW_PASS
- USW_REMOTE_ID
- USW_ID
- USW_PORT
- UVLAN
- SWITCH_IP
- SWITCH_PORT

See **Firewall rules: Variables** on page **187** for details.

One can also add other DHCP options in the switch properties. These options and their values will be included in the DHCP response if a DHCP client includes those options in the DHCP request.

1. **DHCP pool**- an entity that contains an IP pool parameters such as standard DHCP options that are used to form the DHCP response when providing an IP address from this IP pool. The mandatory options are:
   - **Mask**
   - **DNS server 1**
   - **Lease time –** is the lease time in seconds (lease time less than 3600 seconds is not recommended). The default value is 86400 seconds (which is 24 hours).

The nonmandatory options are:

  - **DNS server 2**
  - **NTP server**
  - **Domain**

One can add extra **DHCP options** to the DHCP pool properties. These options will be added to the DHCP response if the DHCP client adds them to the list of the requested options.

The Gateway and Mask parameters are used to identify which DHCP pool does an IP address belong to.

IP address ranges are also a part of a DHCP pool properties. Every IP address range is defined by the first and the last IP address.

Besides that there's a **Block action type** for every DHCP pool. This parameter determines how a DHCP request from a blocked user is processed. It allows one to provide IP addresses from this pool only to blocked users, provide IP addresses from this pool despite of user being or not being blocked, or to ignore requests coming from blocked users.

If certain DHCP options are specified in a DHCP request, the DHCP response will include those options if their values are set in the database. If no options are specified in a DHCP request, the DHCP response will include all the options whose values are set in the database.

1. **IP group** - is a description of a network and its parameters, associated with an IP traffic service link. The IP traffic is identified for the following tariffing and IP addresses are provided based on the IP group settings. An IP group defines the link between a static IP address or a dynamic IP address pool and the following parameters:
   - MAC address
   - Internal switch ID
   - Switch port
   - VLAN ID

One needs to set a static IP address or a dynamic IP pool and set the values for the parameters named above to define that link.

The static IP address ranges shouldn't cross the dynamic IP address ranges. One shouldn't use a static IP address pool in one IP group and as a dynamic IP pool in another IP group. That may lead to an inappropriate UTM5 DHCP behavior.

Port and VLAN ID are DHCP option 82 parameters. UTM5 DHCP reads them according to the **Switch type** chosen for the current switch.

After adding an IP traffic service link and adding an IP group containing parameters defining a link with an IP address, a record of correspondence of a static IP address or a dynamic pool and those parameters will appear in the database.

Processing a DHCP request

When receiving a DHCP request, UTM5 DCHP compares the parameters of the request with the IP group's parameters from the database. The parameters priority is as follows: MAC address > Switch ID > Port > VLAN ID. The database is sorted descending by these parameters and then the search is performed. Each database record contains parameters for a single IP group.

UTM5 DHCP server only compares database records that contain one of the following sets of parameters:

- MAC address
- MAC address and Switch ID
- MAC address, Switch ID and Port
- MAC address, Switch ID, Port and VLAN ID
- Switch ID
- Switch ID and Port
- Switch ID, Port and VLAN ID

1. MAC address is compared first (if it is set in the database)
2. Then the DHCP server reads the option 82 parameters (if it is present in the DHCP request), based on the appropriate **Switch type** parameters for the switch specified in the database record that is currently being checked.
3. If the parameters were read correctly, UTM5 DHCP server compares those parameters with the corresponding parameters present in the database.
4. If one of the parameters is not present in the DHCP request, it is ignored for the comparison.
5. The comparison is considered to be successful if there are DHCP request parameters that match the corresponding parameters from the database and there are no corresponding parameters that do not match.

After providing an IP address, UTM5 DHCP server adds a record, containing the IP address lease start date and the lease time (lease time is set in the DHCP pool properties) to the database.

Configuration file

The default configuration file for unix-systems is `/-netup/utm5/dhcpd5.cfg` and in windows builds it is `dhcpd5.cfg` which can be found in the installation directory (by default it is `C:\Program Files\NetUP\UTM5\`).

Configuration file format:

```
parameter=value
```

A sequence of symbols before the equals sign is treated as parameter's name, while the one after it stands for the parameter's value. Whitespaces count. Empty lines are ignored. Any line starting with # is considered to be a comment.

The list of the possible parameters:

Parameter	**Possible values**	**The default value**	**Description**
`database_type`	mysql, postgres (mandatory parameter)	mysql	UTM5 database type
`database`	(mandatory parameter)	UTM5	UTM5 database name
`database_host`	Database server IP/hostname	localhost	UTM5 database host name
`database_login`	string	root	Login name to access the UTM5 database
`database_password`	string	not set	Password use to access the UTM5 database
database_sock_path	file path	/var/run/mysqld/mysqld.sock	MySQL only. The path to the database socket file. This parameter is used if the database_host parameter is not set or has a "localhost" value.
database_port	1 to 65534	3306	MySQL only. The database server port
`database_charset`	encoding	utf8	MySQL only. Database connection encoding
`core_host`	IP address (mandatory parameter)	127.0.0.1	UTM5 core host
`core_port`	1 to 65534 (mandatory parameter)	12758	Port which UTM5 listens for Stream messages (`stream_bind_port` parameter in the core's configuration file)
`dhcp_login`	string	dhcp	Login name for UTM5 core access
`dhcp_password`	string	dhcp	Password for UTM5 core access

Parameter	**Possible values**	**The default value**	**Description**
`interface`	<interface name>,<IP address> - pairs of parameters, separated by comma. Add multiple interfaces each at new string: interface=<interface name>,<IP address> interface=<interface name>,<IP address> etc...	not set	List of interfaces, accepting DHCP requests and corresponding IP addresses. If the name of the interface is eth0, there are two possible cases: 1) IP address is set (e.g. 10.0.0.1) - the server accepts only unicast requests to address 10.0.0.1:67 2) IP address is set to 0.0.0.0 - the server receives broadcast requests, that come to eth0 (in Linux socket options SO_BROADCAST and SO_BINDTODEVICE are used)
is_authoritative	yes, on, enable	disabled	DHCP server authoritative or not authoritative mode
load_log	yes, on, enable	disabled	Load leases log at DHCP startup. The default value should be good for most systems
`log_level`	number 0 to 3	1	Determines the level of the messages that get to the main message stream
`log_file_main`	file path	standard error flow	The main message flow log file
`log_file_debug`	file path	standard error flow	The debug message flow log file
`log_file_critical`	file path	standard error flow	The critical error flow log file
`rotate_logs`	yes, on, enable	rotation disabled	Enables the log file rotation
`max_logfile_count`	number	not limited	Maximum number of stored log files
`max_logfile_size`	size in bytes	10485760	The maximum log file size after which the rotation happens.
ping_retry_count	number	1	The ICMP request retries limit
use_ping	yes, on, enable	disabled	If an existing lease is found when trying to give an IP address to a client, send an ICMP request to that IP address to find out the actual status of the client
use_old_lease	yes, on, enable	disabled	Renew lease for a particular MAC address in case the DHCP option 82 parameters can’t be matched

22

IPTV INTEGRATION MODULE

Introduction

The IPTV integration module is intended for integration with an IPTV system. When using it with NetUP IPTV system, this module allows one to grant access to IPTV contents and content groups and choose the content explicitly when creating a service. It also allows one to create IPTV access cards and generate activation codes for those cards. When using this module with some other IPTV system, one has to use UTM5 RFW events and third party scripts to control user's access to IPTV contents.

This module requires a separate license. To verify the availability of the license and its terms of validity, see **About: Licenses** on page **106** in the UTM5 administrator's interface and check the list item denoted **IPTV module**.

Common features

The integration module interacts with IPTV middleware. When UTM5 needs to grant access to media content, e.g. when an IPTV service is attached to client's personal account, it sends a request to IPTV middleware to grant access to certain media content to access card owner for an unlimited period of time. When it needs to prevent a user from accessing the content (e.g. when an IPTV service is unattached from user's personal account or when user switches tariff plan), UTM5 requests from IPTV middleware to set access end time for user's access card to that media content to current time, which means that the content becomes no longer available.

In order to let the module connect to NetUP IPTV Core, the DNS server on the UTM5 server should be able to resolve IPTV domain names.

Integration with IPTV systems other than NetUP is possible, but it requires using UTM5 RFW events and it depends on features of each particular IPTV system. The process of UTM5 setup for integration with NetUP IPTV system is described below.

Buying the module license makes IPTV service type available.

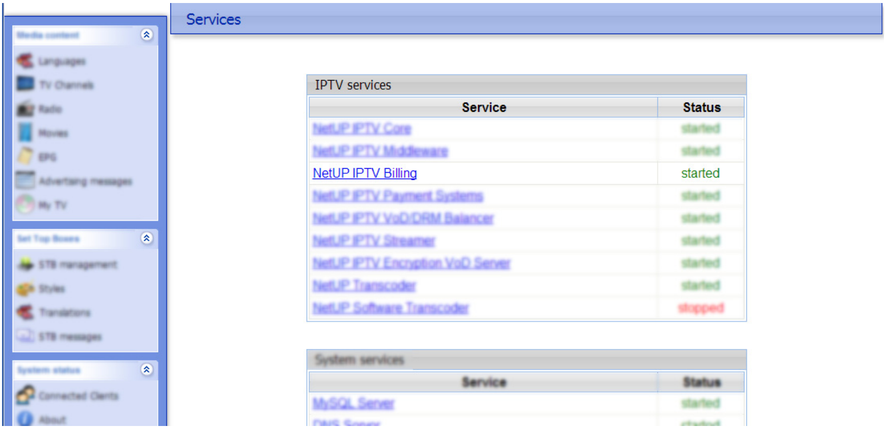
Setup

In order to connect to NetUP IPTV Core, open UTM5 configuration file and set the following parameters:

- **iptv_cluster_host –** is the IP address of NetUP IPTV Core

- **iptv_cluster_port –** is the port which NetUP IPTV Core is listening to for billing system connections

  Open IPTV administrator's web interface and go to the *Services* page:



Services

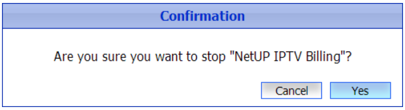
IPTV services

Service	Status
NetUP IPTV Core	started
NetUP IPTV Middleware	started
NetUP IPTV Billing	started
NetUP IPTV Payment Systems	started
NetUP IPTV VoD/DRM Balancer	started
NetUP IPTV Streamer	started
NetUP IPTV Encryption VoD Server	started
NetUP Transcoder	started
NetUP Software Transcoder	stopped

System services

Service	Status
MySQL Server	started

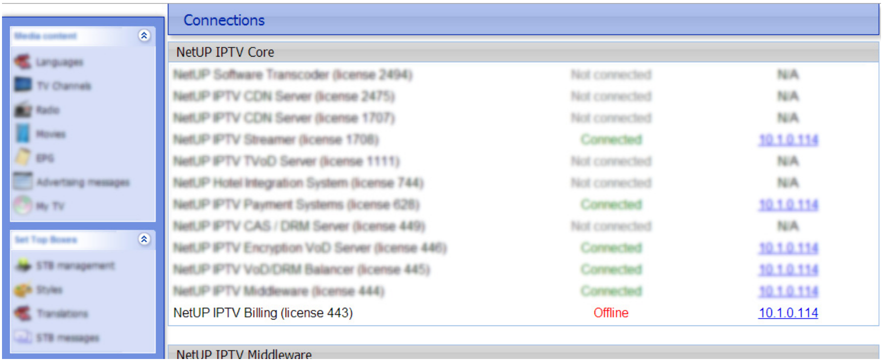
If *NetUP IPTV Billing* service is running, left-click on it’s name to open a dialog window. Press **Yes** when that window appears to stop the service.



Confirmation

Are you sure you want to stop "NetUP IPTV Billing"?

After that go to the *Connections* page and left-click the IP address next to the *NetUP IPTV Billing* connection name.



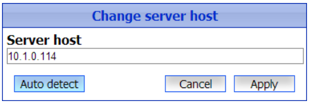
Connections

NetUP IPTV Core

NetUP Software Transcoder (license 2494)	Not connected	N/A
NetUP IPTV CDN Server (license 2475)	Not connected	N/A
NetUP IPTV CDN Server (license 1707)	Not connected	N/A
NetUP IPTV Streamer (license 1708)	Connected	10.1.0.114
NetUP IPTV TVoD Server (license 1111)	Not connected	N/A
NetUP Hotel Integration System (license 744)	Not connected	N/A
NetUP IPTV Payment Systems (license 628)	Connected	10.1.0.114
NetUP IPTV CAS / DRM Server (license 449)	Not connected	N/A
NetUP IPTV Encryption VoD Server (license 446)	Connected	10.1.0.114
NetUP IPTV VoD/DRM Balancer (license 445)	Connected	10.1.0.114
NetUP IPTV Middleware (license 444)	Connected	10.1.0.114
NetUP IPTV Billing (license 443)	Offline	10.1.0.114

NetUP IPTV Middleware

In the *Change Server Host* window that will open, press **Auto detect** to reset the information about the billing system.



Change server host

Server host: 10.1.0.114

Now launch UTM5. After that go to the *Connections* page of the IPTV administrator's web interface and make sure that NetUP IPTV Billing is connected and the IP address matches the IP address of the server that is running UTM5.

Creating and attaching IPTV service

First one needs to create an IPTV access card:

1. Open UTM5 administrator's interface and go to the Users page
2. Choose a user and press Edit
3. Once the User properties window opens, go to the Accounts page in the Tarrification group
4. Choose an account to which you plan to attach IPTV service and press Edit
5. Once Account properties window opens, press Create button next to the IPTV access card parameter. An access card number will appear in the parameter field.

One can find an activation code for the access card on IPTV activation codes page in Tariffication group.

Now create a new service (for more information see **Usage examples: Creating services** on page **132**). Go to the Service parameters page choose NetUP as an IPTV system and choose a media content or a group of media contents which a user, to whose account this service will be attached, should be provided access to.

After that one can attach this service to a user's account like any other periodic service (see **Usage examples: Creating service links** on page **136**).

Create an IPTV service template if you need to include this service into a tariff plan.

23

IP TELEPHONY MODULE

Introduction

The IP telephony module is intended for processing authorization requests and consumed services accounting for voice gateways, gatekeepers and voice proxy servers. It supports both traditional and IP telephony. The data to be accounted for may be based either on RADIUS server requests (see **UTM5 RADIUS** on page **159**) or on CDR files parsed by the `utm5_send_cdr` utility (see **Text files import** on page **179**).

The IP telephony module requires a separate license. To verify the availability of the license and its terms of validity, see About: Licenses in the UTM5 administrator's interface and check the list item denoted **Telephony module**.

List of terms

- **IP telephony –** is a general term denoting voice transmission over networks via IP. Also known as: Voice over IP, VoIP, Internet Telephony.
- **PSTN –** is a Public Switched Telephone Network. This notion includes local and national telephone networks.
- **Caller ID –** is a phone number of a caller.
- **ANI –** is Automatic Number Identification.
- **VoIP gateway –** is a device with an IP port and also (if required) ports to connect to PSTN. Usually the device is used as a gateway between PSTN and IP network. Cisco router 3620 with the NM-2V + VIC2FXO module may serve as an example of a device of this type.
- **H.323 –** is a standard offered by the International Telecommunications Union (ITU-T) describing principles of IP telephony networks. The standard describes the protocols associated with IP telephony equipment registration (RAS – Registration, Admission and Status), call setting-up (H.225.0, H.245), voice transmission, user authorization, etc.
- **H.323 gatekeeper –** is responsible for registration of terminal equipment (gateways, client devices), access rights control, distribution of numbers. Almost all gatekeepers can process authorization and transmit statistics on telephone calls via RADIUS protocol.
- **Codecs –** are the sound compression algorithms on the transmission side and decompression on the receiving side. Generally are used to minimize network traffic. That's why codecs are usually characterized by the bandwidth required for voice transmission using this codec. Uncompressed voice transmission takes 64 Kb per second.

Codecs with high compression ratio require powerful computing resources. That's why encoding of a large number of voice flows requires usage of special microprocessors (DSP, digital signal processor).

Codec	Bit rate, Kbit/sec	Quality
G.711	64	High
G.723.1	5.3 – 6.4	Medium
G.729	8	Medium

- **IVR –** is Interactive Voice Response. Represents a technology of voice menu and is widely used for authorization of PSTN users to call via IP telephony.

Workflow description

RADIUS requests concerned with telephony are recognized by the system based on the cisco-h323-conf-id attribute. If it is missing, the request is interpreted as related to dialup service.

To register a caller, a gateway sends to the RADIUS server a registration request containing the Calling-ID (31) and caller’s login, but no Called-ID (30). The RADIUS server in turn searches for the telephony service link which is identified by the login in its properties (see **Telephony service link: Login** on page **115**). If the link in question is not found or the corresponding account appears to be blocked, the registration is denied. Otherwise, an affirmative response is sent, which may also contain the user’s phone number if it is set in the service link properties.

To authorize a call, a gateway or a voice gateway sends to the RADIUS server a registration request containing the Calling-ID (31) together with Called-ID (30). The RADIUS server in turn searches for the telephony service link which is identified by the login in its properties. If the link is not found, or the account is blocked, or the call parameters do not match those of any direction covered by the service, or the current time is not covered by the service, the registration is denied. Otherwise, an affirmative response is sent, which also contains the maximum duration of a connection. The maximum duration is calculated either as the time left till the end of time range covered by the service (unless the service provides round-the-clock coverage), or as the time till the account’s balance runs out given its current balance and the current connection price per minute (which may also be time-dependent), whichever of these happens sooner.

To account for a call, a gateway or a voice gateway sends to the RADIUS server an Accounting-Start request containing the Calling-ID (31), Called-ID (30), and probably the starting time. If the starting time is not provided, the arrival time of the Accounting-Start request is assumed instead. The RADIUS server in turn searches for the telephony service link which is identified by the login in its properties. If the link is not found, the request is ignored. Otherwise the connection price per minute is determined, which may depend on the telephone direction and current time. If the call parameters do not match those of any direction covered by the service, or the current time is not covered by the service, the call is accounted for by zero price. When a call finishes, an Accounting-Stop request is sent containing the Calling-ID (31), Called-ID (30), and probably the call duration and/or its finishing time. Then the RADIUS server accounts for the call considering its duration and the price per minute. If the call duration is not provided, the difference between the finishing time and starting time is assumed instead. If the finishing time is not provided, the arrival time of the Accounting-Stop request is assumed instead. If the price per minute is time-dependent and does change during the time span in question, the call is split into parts of constant price and accounted for in parts. The charge-off information is passed to the UTM5 core.

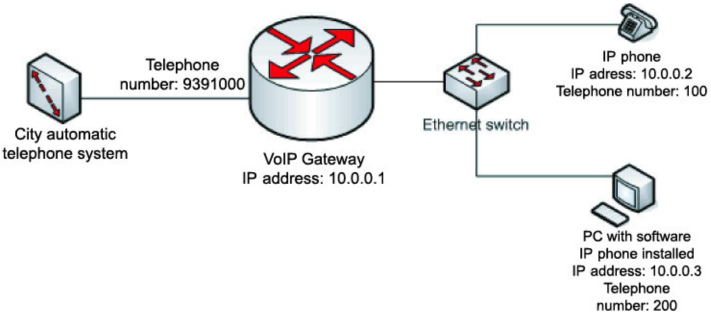
The calls lacking an Accounting-Stop request may be either ignored or considered finished by timeout based on Interim-Update requests and accounted for accordingly, depending on the RADIUS server settings.

If the gateway does not support the Accounting-Request communication with the RADIUS server, it may dump the phone call information to text files to be parsed later by the `utm5_send_cdr` utility (see **Text files import** on page **179**). This utility parses log files, retrieves individual calls and sends those to the UTM5 core using URFA.

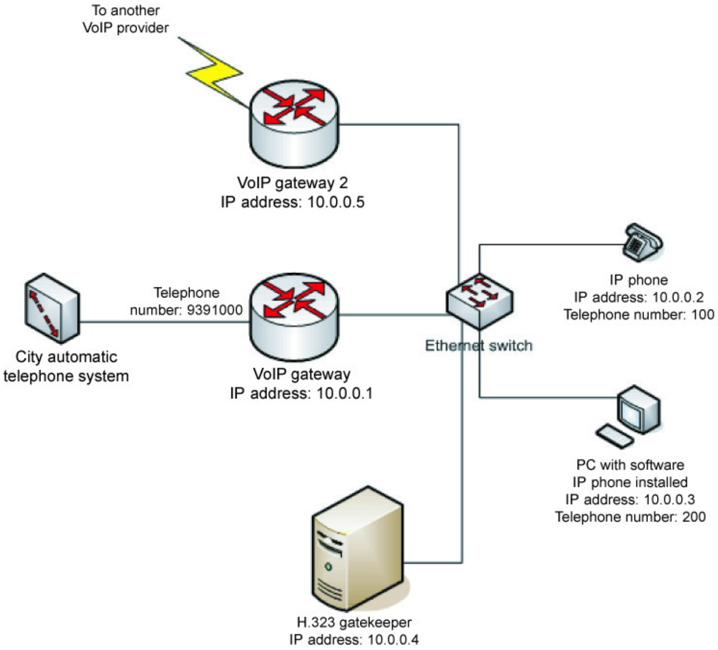
Network organization schemes

A VoIP gateway connecting PSTN to an IP network organizes voice traffic conversion from IP network to PSTN. Thus, a user with an IP phone or a PC with a software phone installed (Microsoft NetMeeting, OpenPhone etc.) may give a call to a subscriber of PSTN.

Similarly a subscriber of PSTN may call a network user. For that it is required to dial the gateway phone number in PSTN (9391000 on the scheme) and then, after authorization (if the mechanism is enabled on the gateway) dial an internal number of an IP network user (numbers 100 and 200 on the scheme).



In the scheme containing the H.323 gatekeeper, all devices should register on the gatekeeper. At that, authorization may be processed via RADIUS protocol by using the common Access-Request scheme.



As a result the gatekeeper has a table with IP addresses and numbers of all network devices. All calls begin with addressing to the gatekeeper for conversion of dialed numbers to IP addresses. For that the gatekeeper requests of the RADIUS server to authorize the call and pass the filled in attributes Called-Station-Id (30) (dialed number) and Calling-Station-Id (31) (a number of a calling subscriber). At that the RADIUS server checks a user balance, tariff plan for a called direction and, if all is OK, gives the Access-Accept packet in which it may set the maximum connection duration for the user calling to the certain direction. Usually this information is set in the h323-credit-time, vendor 9 attribute (Cisco).

In case authorization is successful (and after all parameters are coordinated) the connection between a called and a calling station is established. At that the gatekeeper sends a packet (Accounting-Start) containing parameters for the established connection to the RADIUS server.

In case both stations are in the same network the connection is being established directly. If the called station is in another network then the connection is established via one of the gateways. Another variant is also possible, when a user communicates with the gatekeeper only. In this case the gatekeeper acts as a proxy server and real IP addresses of the stations are hidden. This scheme may be applied if the direct line between the stations is worse (e.g., serious IP packets loss or a delay) than between the gatekeeper and both of the stations.

When the connection is finished the gatekeeper sends a packet containing information about the call to the RADIUS server. In the packet it specifies the connection duration, a cause of the connection break and other parameters. Using these data the RADIUS server rates the session, charges the user and puts a record in a log file.

Authorization of PSTN users may be done using IVR as follows:

1. A user of PSTN dials a local number of IP telephony access. The call is accepted by an IP telephony gateway (e.g., Cisco 3640 with E1 module) connected to the line.
2. The gateway loads an audio file (usually of the `.au` type) with an invitation record and plays it to the user. Usually it prompts the user to enter a number and a PIN code of a prepaid telephone card.
3. After a special digit combination is entered, the authorization is being processed on the RADIUS server. At that, the card number and PIN code are usually recorded to the attributes 1 (User-Name) and 2 (Password).
4. In case of successful authorization the RADIUS server sends an Access-Accept packet with the user balance. For that the attributes h323-credit-amount and h323-currency with vendor=9 (Cisco) are used. IP telephony gateway loads appropriate voice files and in this way informs the user of his balance and invites to enter a telephone number. Note that usually IP telephony is profitable for remote calls (national and international calls).

5. After the number is entered it is processed through second authorization on the RADIUS server. At that, an attribute Called-Station-Id containing the dialed number is transmitted additionally. Depending on the balance and connection cost per minute, the RADIUS server calculates the maximum available session duration and sends the value in the Access-Accept packet attribute h323-credit-time.
   If the Called-Station-Id attribute is missing, the R-ADIUS server returns h323-return-code (9,103) attribute with the following meaning:
   - 0 means that the user is active;
   - 1 means that the user does not exist;
   - 2 means that the user is blocked.
1. After the affirmative reply is received from the RADIUS server, the IP telephony gateway establishes connection with the called user. The connection will break if the session duration exceeds the maximum calculated in the previous step.
2. On establishing the connection an Accounting-Start packet is sent on the RADIUS server. On breaking, the Accounting-Stop packet is sent.

24

AUTOMATIC REGISTRATION OF USERS

Introduction

UTM has two options for activating the prepaid Internet cards and receiving the dial-up service: guest access or conventional access with automatic registration of users. In the first case, the guest login and password are used to register in the system. After registration the user enters the system with one’s own access parameters. In the second case the user enters card number and a pin code as a login and password for dial-up connection, and then is automatically registered and gets access to the Internet right at once.

For automatic registration of users using the options above, you have to generate the tariff plan and connect the dial-up service to the user with the corresponding connection cost. On creating the tariff plan you have to generate the prepaid cards pool and bind it to the tariff plan.

Guest access

If you use guest access you have to generate a user with a login and a password known beforehand. For example, login "guest" and password "guest".

These settings should allow the guest access only to the web-site to activate the Internet-card. The session time can be also restricted, say, to 600 seconds.

It is necessary to create a **Dial-up access** service with pool GUEST, maximum connection timeout of 600 seconds, and connection cost of 0 c.u. (currency units) per hour.

Then you have to generate a pool of IP addresses with login GUEST and certain-range addresses, e.g., 172.16.0.0/16, on the router or in UTM. The router settings should allow this range of users to get access only to the DNS and to the web server to activate the card. For the safety sake it is better to arrange an isolated DNS server, not connected to the Internet and containing only the records, which the user will need to access the registration web server.

On login the registration web server the user selects the "Automatic registration of user" menu item and enters the data from the Internet-card. If the data is entered correctly and the card was not activated or blocked in the past, a new card user will be generated in UTM automatically, and the user will receive the login and password info for connection in dial-up mode. By selecting the "Login to UTM" menu item and entering the login and the password received after the registration the user may get access to his personal office and account statistics.

Access with automatic registration

The immediate access with the prepaid cards requires RADIUS server additional tuning. In RADIUS server configuration file `/netup/utm5/radius5.cfg` define the option:

```
radius_card_autoadd=yes
```

Restart the server. RADIUS server will automatically register the user in UTM at his first attempt to get access on prepaid card.

In order to receive access the user should enter the card number as login and its pin code as a password on every connection. If the user connects with this card at a first time RADIUS server will register him automatically and connect him to the Internet right at once. When the card expires (balance turns red), the user has to activate a new card.

This kind of automatic registration is possible on authorization using PAP protocol only. This method is used by Windows by default for modem connections and requires no additional settings. However, sometimes the users' configuration should be changed to let them be registered automatically.

If access settings for the user automatic registration are correct the following records should appear in the RADIUS server log file on the first connection:

```
?Debug: Oct 27 12:08:00 RADIUS Auth: Packet from <example.org>
?Debug: Oct 27 12:08:00 RADIUS Auth: User <5> connecting
ERROR: Oct 27 12:08:00 RADIUS DBA: Can't find login <5>
ERROR: Oct 27 12:08:00 RADIUS DBA: Can't find card login
<000000005>
?Debug: Oct 27 12:08:00 RADIUS Auth: Attempt to add new Card
user: <5>
?Debug: Oct 27 12:08:00 RADIUS DBA: Sending Auto-Add Request for
Card-ID: 5
?Debug: Oct 27 12:08:00 RADIUS URFA[plugin]: DLink:
SLID/SID/AID: 14/6/14
?Debug: Oct 27 12:08:00 RADIUS URFA[plugin]: Account <14> with
balance <10.000>
?Debug: Oct 27 12:08:00 RADIUS Auth: Got AutoAdd 14 UID from
core.
ERROR: Oct 27 12:08:00 RADIUS DBA: Can't find login <5>
?Debug: Oct 27 12:08:00 RADIUS DBA: login_store iter-
>second.dialup.session_count:0
Info: Oct 27 12:08:00 RADIUS Auth: User <5> added.
?Debug: Oct 27 12:08:00 RADIUS Auth: Auth scheme: PAP
```

```
?Debug: Oct 27 12:08:00 RADIUS Auth: PAP: <51154755> vs
<51154755>
?Debug: Oct 27 12:08:00 RADIUS Auth: PAP: Authorized user <5>
?Debug: Oct 27 12:08:00 RADIUS Auth: Dialup session limit:0
session count:0 for user:5
?Debug: Oct 27 12:08:00 RADIUS Auth: Calculated maximum session
time: 36000
?Debug: Oct 27 12:08:00 RADIUS DBA: dialup_link_up-date called
for slink:14
?Debug: Oct 27 12:08:00 RADIUS DBA: soft dialup_ link_update for
slink:14 session_count:1
```

E-MAIL NOTIFICATIONS

25

UTM5 may send automatic e-mail messages to the users (or rather those of them who has valid e-mail addresses entered in their user info) for a number of reasons. Global system parameters related to e-mail are described in **UTM5 core: Interface parameters** on page **152**.

The messages are sent via SMTP server set by the `smtp_relay` parameter. The SMTP server must be set up correctly and must send every incoming message within 1 second. Longer delays in email processing may drastically reduce the billing performance. It is recommended to use the local SMTP server.

Possible types of e-mail messages include:

- **Invoices –** are sent when either an invoice is issued to a user having the **Send invoices by email** parameter checked (see User: Contacts on page 41), or when the **Send by email** context menu item is hit in the report on invoices (see **Report on invoices** on page **79**). Message subject is set by the `invoice_-subject` system parameter. Message text is set by `invoice_text`, while the invoice itself is contained in an attachment as an HTML file. The invoice is generated on the basis of the **Invoice** document template (see Document templates on page 83).
- **Payment notifications –** are sent on the event of a payment being made, if the corresponding parameter is checked in the payment's properties (see **Payment page: Send email notification** on page **107**). Message subject is **Payment nofication**. Message text is composed from the template stored in the `payment_notification_message` system parameter.
- **Balance notifications –** are sent when the user's balance (not considering the credit) crosses the borders defined by the `notification_borders` system parameter, if the latest is set. Message subject is defined by the `notification_message_-subject` system parameter. Message text is composed from the template stored in the `notification_message` system parameter.

SYSTEM MAINTENANCE

26

Database backup

To prevent possible loss of data, it is recommended to make backup copies of the database regularly (say, monthly). This is normally done with the standard tools specific to the particular kind of DB server. Besides regular backups, it is also advisable to make an extra copy before any low-level operation on the database, like archiving of tables, direct manual intervention, debugging of urfaclient scripts, etc.

The backup copy may be either brief or full. The latter one contains all tables, while the former one omits the charge-off tables. It is recommended to stop the UTM5 core while creating a full backup copy (which may take considerable time, due to the excessive size of charge-off tables). Otherwise prolonged blocking of tables may lead to core crash.

For large projects, where the tables are especially huge and yet it is critical to keep downtime low, we recommend the use of a slave DB server, which makes it possible to create a backup copy without shutting down the billing.

Database integrity verification

Once the UTM5 core is started, it fills the system cache and verifies the database. The revealed inconsistencies in the cached data are resolved automatically. However, the original data in the database remain corrupted and have to be fixed manually. To do that, one may use the verifier log file.

The location of the said file is given by the `log_file_v-erificator` system parameter (by default, `/netup/log/verificator.log`). For each item it contains:

- Description of the inconsistency, including its level (ERROR or WARNING);
- Supposed way to resolve the issue;
- SQL command (if required) equivalent to the automatic fix applied to the cached data:

```
-- WARNING slink 4876 exists only in dtagg_periodic
-- SQL DESC check slink exists and delete dtagg_periodic entry
for deleted slink
UPDATE dtagg_periodic SET is_closed=1 WHERE slink_id=4876;
```

The objects listed in `verificator.log` *as condemned to deletion are not loaded by the system and also neither accounted for in the reports nor shown wherever in the administrator’s interface.*

When applying the fixes to the database, it is desirable to stop the UTM5 core and create a backup copy of the entire database, or at least of the tables affected by the fix.

In the trivial case all fixes may be applied by simply feeding the verifier log file into MySQL:

```
mysql UTM5 < /netup/utm5/log/verificator.log
```

However, some SQL queries in the log file are commented out, since they imply some (probably undesired) loss of data. When dealing with such queries, one has to check every individual issue separately.

Archiving of tables

Some of the fastest-growing UTM5 data tables may be archived in order to reduce the overhead expenses on insert operations. An archiving implies that the table in question is renamed into an archive table, while an empty table with the original name and structure is created to store the incoming data. Archiving may be done periodically. The limitations are listed below.

Currently the following tables are being archived:

Table	**Type**	**Date field name**
`discount_transactions_all`	1	`discount_date`
`discount_transactions_ ip-traffic_all`	2	`discount_date`
`tel_sessions_log`	3	`recv_date`
`tel_sessions_detail`	4	
`dhs_sessions_log`	5	`recv_date`
`dhs_sessions_detail`	6	
`payment_transactions`	7	`payment_enter_date`
user_log	8	date
dhcp_leases_log	9	updated
invoices	10	invoice_date
invoice_entry	11	
invoice_entry_details	12	

In order to archive these tables:

1. Use the administrator’s interface to connect to UTM5

2. Go to the *Archive DB* page in *Settings* group of pages
3. Press Create in the upper part of the page to create an archive

One can do archiving once in 28 days. If Create button is not active, this means that less than 28 days passed since the last archive was created.

Should you need to do the archiving more often, than is allowed by the administrator’s interface, please use the *db_archiver* utility (see **db_archiver utility** on page **275**)

AUXILIARY UTILITIES

NetFlow statistics generator

To emulate activity of users and export statistics via NetFlow v.5 protocol there is a utility called `utm5_flowgen` which is installed to `/netup/utm5/bin/utm5_flowgen`. It may accept the following command line parameters:

`-h`	IP address of the host to send generated NetFlow packets to. Default value is 127.0.0.1
`-p`	Port to send generated NetFlow packets to. Default value is 9996
`-c`	Number of NetFlow records. Default value is 65535
-v	NetFlow protocol version. Supports versions 5 and 9
-f	Name of a file which will be used as the source of data for sending. The default source is `/dev/random/`
`-t`	Delay between sendings of NetFlow packets (in microseconds)
`-s`	Sender IP address in the NetFlow record
`-d`	Destination IP address in the NetFlow record
-z	Traffic source port in the NetFlow record
-x	Traffic destination port in the NetFlow record
-n	Traffic source AS in the NetFlow record
-m	Traffic destination AS in the NetFlow record
-i	Incoming traffic index in the NetFlow record
-o	Outgoing traffic index in the NetFlow record
`-b`	Number of transmitted bytes in the NetFlow record
-P	Number of transmitted packets in the NetFlow record
-j	TOS in the NetFlow record
-k	TCP flags in the NetFlow record
-l	Protocol ID in the NetFlow record. E.g. 6=TCP, 17=UDP, etc.
-N	Next router IP address in the NetFlow record
-u	Use a *.utm file as a source for the detailed NetFlow statistics

The following example command generates one NetFlow packet describing 1048576 bytes of traffic transmitted from 10.0.0.1 to 10.0.0.2:

```
/netup/utm5/bin/utm5_flowgen -c 1 -s 10.0.0.1 -d 10.0.0.2 -b
    1048576
```

RADIUS statistics generator

For emulation of user activity and export of statistics via RADIUS protocol there is a utility called `utm5_radgen` which is installed to `/netup/utm5/bin/utm5_radgen`. It may accept the following command line parameters:

`-p`	Port for generated RADUIS packets to be sent to
`-h`	IP address for generated RADIUS packets to be sent to
`-s`	Secret word for communicating with RADIUS server
`-c`	RADIUS packet code. Default value is 1 (Access-Request)
-i	RADIUS packet ID. Default value is 1
`-u`	User password in public form. The value is sent with attribute ID equal to 2 (Password)
`-a`	Attribute values
-b	Binary attribute values in HEX ASCII
-q	Quick mode - don't wait for reply
-f	Name of a file to read the authenticator from. Default value is `/dev/random`
-v	Display utility version

It is possible to set multiple attributes in a string of the following format:

```
vendor_id:attr_id:is_digit:value
```

Fields are separated by colons. In the first field the vendor identifier is set. Default value is 0. The second field contains attribute identifier. The third field is used to set data type, i.e. numeric or char. If the value is 0 then the data is transmitted as a character string. If the value is 1 then values are transmitted as digits (integer). The 4th field is used for transmission of the value itself.

Examples

1. To send an authorization request (Access-request) run the following command:

```
/netup/utm5/bin/utm5_radgen -h 127.0.0.1 -p 1812 -s secret -u
password -a 0:1:0:username
```

A RADIUS authorization packet will be generated for a user `username` with password: `password`.

2. To send a request for accounting (Accounting-request) run the command:

```
/netup/utm5/bin/utm5_radgen -h 127.0.0.1 -p 1813 -s secret -a
0:1:0:username -a 0:40:1:1 -a 0:44:0:sessionid1 -c 4
```

A RADIUS packet will be generated with the accounting request for a user `username` stating that a session with identifier `sessionid1` is being started.

3. To send a request for accounting (Accounting-request) run the command:

```
/netup/utm5/bin/utm5_radgen -h 127.0.0.1 -p 1813 -s secret -a
0:1:0:username -a 0:32:0:localhost -a 0:40:1:2 -a
0:44:0:sessionid1 -a 0:46:1:100 -c 4
```

A RADIUS packet will be generated with the accounting request for a user `username` stating that the session with identifier `sessionid1` is being stopped. Session duration (Acct-Session-Time) is 100 seconds.

get_nf_direct utility

The `get_nf_direct` utility is designed to form detailed traffic reports based on the saved raw information.

The executable file is called `/netup/utm5/bin/get_nf_direct`.

Possible command line parameters are:

`-D <dir>`	Path to directory containing the primary traffic information files
`-b <database filename>`	Name of file with primary traffic information
`-a`	Account ID for the report
`-s <source ad-dress>`	Traffic source ID for the report
`-d <destination address>`	Traffic destination ID for the report
`-p <source port>`	Source port for the report
`-P <destination port>`	Destination port for the report
`-c <t_class>`	Traffic class for the report
`-f <from t-imestamp>`	Time (Unix timestamp) to create the report since
`-t <to time-stamp>`	Time (Unix timestamp) to create the report till (if not set, current time is used)
`-l <limit>`	Maximum number of lines in the report (unlimited by default)
`-e`	Represent extended statistics
-C	CSV format output
`-h`	Version and usage info

db_archiver utility

db_archiver is used when updating UTM5. It allows one to compare the current DB structure with the one needed by the updated UTM5 core. It allows one to update the DB structure and to archive tables that are meant to be archived.

The executable file is `/netup/utm5/bin/db_archiver`.

Command line parameters:

`-a`	Archive tables that are meant to be archived
`-c <path>`	UTM5 configuration file path. By default: `/netup/utm5/utm5.cfg`
`-d`	Write the difference between the current DB structure and the DB structure, required by the new UTM5 core to the log file
`-e`	Update only those columns that have changed since the previous release and are marked for update by NetUP
`-f`	Update all columns whose format is different from the format, required by the new version of the UTM5 core
`-g`	Update the structure of tables meant for archiving
`-i`	Update indexes
`-n`	For MySQL do not consider a primary key without a default value a difference. For PostgreSQL do not consider a primary key with NOT NULL constraint a difference
-t	Verify archived tables
`-l`	Temporarily prevent the UTM5 core from writing to the DB. This is required when archiving tables without stopping the core. Use this option together with -a when the core is running
`-q`	Turn off confirmations and minimize the output to log file. This may be useful when running this utility on a schedule
`-u`	Update the DB structure. Use this option together with -e, -f, -g, -i
`-v`	Write the new DB structure description to the log file
`-x <login>`	Login for communicating with the UTM5 core via the URFA protocol
`-y <password>`	Password required for communicating with the UTM5 core via the URFA protocol. Both login and password may be required if for some reason they differ from the ones in the UTM5 configuration file
-?, -h	Show this help

28

PAYMENT SYSTEMS

Introduction

The external payment systems integration module is compatible with UTM since 5.2.1-008 onwards. Currently the module works only on Linux or FreeB-SD operating systems, uses MySQL database server, and supports a broad list of Russian and international payment systems, including PayPal and Yandex.Money.

Installation

To install the integration module, you must have a server running on FreeBSD (9.x or compatible) or Linux, and a database server MySQL 5.6.x. Prior to installation it is necessary to create a sepatare MySQL database:

```
mysqladmin create Payment_systems
```

Download the installation script from your personal cabinet at **http://www.utm-billing.com**. The script is found under **Payment Systems** and called `n-etup-payment-systems-v2-<system>-5.0-rc1.sh`, where `<system>` is either Linux or FreeBSD.

If a previous version of the integration module is running, stop it.

Start the installation script:

```
#./ n-etup-payment-systems-v2-<system>-5.0-rc1.sh -i
```

To get help on the installation parameters, run:

```
#./ n-etup-payment-systems-v2-<system>-5.0-rc1.sh -h
```

Installation script may accept the following command line parameters:

Short form	Long form	Meaning
`-i`	`--install`	installation mode
`-u`	`--update`	update mode
	`--uninstall`	uninstallation mode
`-p`	`--patch-db`	repair mode
`-b`	`--backup`	backup mode
`-e`	`--extract`	extract to the current directory

Short form	Long form	Meaning
`-h`	`--help`	display help
`-v`	`--version`	display version number

After the installation you must:

1. Set the private key password in the config file (see **Config file** on page **279**) to the value specified in your personal cabinet.
2. Download the `netup.keystore` file from your personal cabinet and put it into `/netup/etc`.
3. Start the server part of the integration module (see **Server part startup** on page **278**).
4. Install and start the control center (see **Control center installation** on page **278**).
5. Set up the parameters of connection to UTM5 (see **Connection with UTM5** on page **285**);
6. Set up the parameters of interaction with the payment systems you are using (see **External payment systems** on page **284**).

For successful installation you must have administrator’s privileges.

Server part startup

The server part of integration module is started as follows.

- For Linux:

```
/etc/init.d/netup-payment-systems-v2 start
```

- For FreeBSD:

```
/usr/local/etc/rc.d/netup-payment-systems-v2.sh start
```

On Linux you might want to add the module to the autorun list. For example, in Debian this is done as follows:

```
update-rc.d netup-payment-systems defaults
```

Control center installation

The integration module is not controlled by the UTM5 control center. Instead, it uses a special control center of its own, which has to be installed separately. To do that:

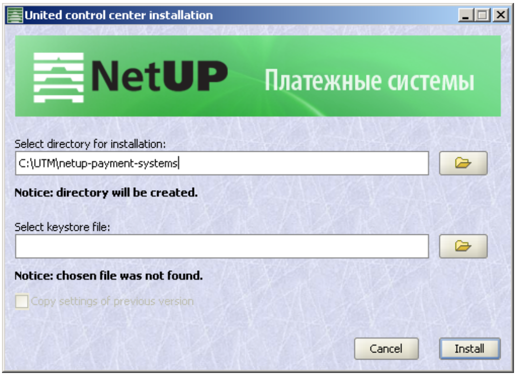
1. In your private cabinet’s **Payment Systems** page download the control center installer called `netup-payment-systems-v2-ucc-install-xxxx.jar` and (if installing for the first time) the `netup.keystore` file.

To install and use the control center, Java Runtime Environment (JRE) version 8.0 (Java 1.8.x) or above is needed.
JRE distributive is available for free at **http://j-ava.com**.

2. Start the installation.

Language of the installer, as well as that of the control center itself, is set according to the current system locale.

3. Select the directory to install the control center to. When installing over the previous version (that is, into the same directory), you may also want to check the **Copy settings** option.



Select directory for installation:

C:\UTM\netup-payment-systems

Notice: directory will be created.

Select keystore file:

Notice: chosen file was not found.

☐ Copy settings of previous version

4. Enter the path to the keystore file (`netup.keystore`).

5. Press **Install** to install the control center to the desired location.

Settings

Config file

The payment systems integration module uses a config file of its own, which is located at `/netup/etc/netup-payment-systems-v2.config.xml`.

The necessary settings include:

- Database connection parameters

```
<database host="your_ip" login="your_db_login" pas-
sword="your_db_passwd" name="payment_systems" />
```

- Private key password

```
<security password="secret" />
```

The rest of the settings include:

- Port settings

```
<transport>
   <xml port="51010" max_connections="10" />
   <https port="8080" ssl_mode="on" priority="3"/>
   <https port="8081" ssl_mode="on" priority="4"/>
</transport>
```

The `ssl_mode` attribute within the HTTP ports descriptions may be one of the following:

- **"off" –** means HTTP connection;
- **"on" –** means HTTPS connection;
- **"certificate verification" –** means HTTPS connection with verification of the SSL certificate (relevant for the payment systems which use it for authentication).

- Logging parameters

```
<logger>
  <appender level="error, debug, warning, info, sql"/>
</logger>
```

The `level` attribute contains comma-separated types of events which should be logged.

- Payment verification command (by default not set, see **External payment systems: Type of check** on page **284**).

```
<tool>
  <filter command="<your_command>" />
</tool>
```

Here `<your_command>` is the full path to the executable file that actually performs the verification.

- Additional parameters (commented out by default):

```
<!--
 <system pid_file="/var/run/netup-payment-systems.pid "
plugins_path="/netup/netup-payment-systems/plugins" t-
imeout="20"/>
 -->
```

When the system is working under high load, you may uncomment this line to adjust the billing timeout period. The default value is 10.

- Default payment currency code:

```
<system default_iso_currency_code="398"/>
```

When changing this, in the first place make sure that the currency with this code does exist in the billing.

- Provider ID for the OSMP payment system:

```
<osmp pay_id="osmp_unique_provider_id"/>
```

- Path to the config file for PayBox (see **PayBox payment system** on page **282**):

```
<sfour configuration="/netup/etc/sfour.xml" />
```

Additional parameters

Working with some payment systems (see the list below) requires some additional parameters which are either passed to the integration module via the tab **Settings: External payment systems** in UCC, or used by external checking utilities.

Payment system	Additional parameters required
Freecash	`SECRET_KEY` (provider's secret word)
Unikassa Terminal PSKB PSKB_EC	`COMPANY_ID` `SECRET_KEY`
Yandex money v.2	ShopID shopPassword [a]
ChronoPay	`SITE_ID`
ComePay	`SECRET` (password)
CyberPlat	x509 certificate (see below)
Handy Bank	`SERVICE_ID` (Handy number + co-number), system's public key, client's private key
Mainpay	Provider's secret word (password)
WebMoney	`MAIN_PURSE` `SECRET_KEY`

a. The shopPassword parameter should be plugged into the **Password** field only if the md5 checking is used, which requires an agreement with Yandex. If this parameter is not entered, the module falls back to the default PGP algorithm.

ID and password

A number of payment systems (see the list above) requires some provider ID (purse number or other similar parameter) and/or the provider's secret word. These parameters are issued to the provider by the payment system. Exact names of the parameters, which may differ for various systems, are also given in the list.

Key exchange

Usage of the Handy Bank payment system implies exchange of the public openssl keys. You have to generate your own pair of keys (public and private) in advance. Then the integration module should be provided with the system's public key (see **Trusted Public key** on the **External payment systems** tab of UCC) and your own private key (**Private key** on the same UCC tab).

Public key

Optionally, the Yandex.Money payment system may issue a public key for GnuPG encryption. In this case it must be imported on the server running the payment systems integration module as follows:

```
gpg --import <public key file>
```

X509 certificate

To use the CyberPlat payment system, you have to produce an x509 certificate from your `netup.keystore` file as follows:

```
openssl pkcs12 -in netup.keystore | openssl x509 -text
```

Mainpay payment system

This payment system works only over HTTP. Hence it is recommended to assign a separate port for it and turn off SSL on that port (by setting `ssl_mode="off"`, see **Config file: Port settings** on page **279**).

Test payments mode of Mainpay is not supported.

PayBox payment system

The PayBox protocol implies that the integration module works with a number of terminals each having its own ID and password. The list of terminals should be stored in a separate XML file of the following format:

```
<terminal MachineMark="No1" ClearingNumber="gr1" Se-cret="pwd1"
State="Blocked"/>
<terminal MachineMark="2" ClearingNumber="gr1" Secret="pwd2" />
```

For each terminal, it should contain its ID, password, group ID (ClearingNumber), and (optionally) its state.

Path to this file should be given in the tag `<sfour/>` of the main config file (see **Config file** on page **279**). When the list of terminals is modified, new settings will be applied automatically without restart.

Logging

Log file of the external payment systems module is stored at `/netup/log/netup-payment-systems-v2.log`.

When the module is reset or receives a SIGHUP signal, the log file is closed, renamed to get the "`.<current timestamp>`" suffix in its name, and a new log file with the original name is started.

Old log files are not removed automatically.

Control center operation

To operate the control center:

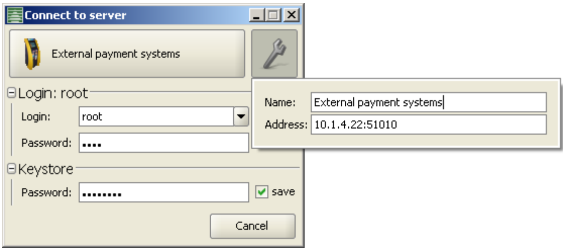
1. Start the control center by running `control.center.se.jar` in one of the two ways:
   - double-click on the file,

   or
   - in the folder containing the file, execute

```
java -jar control.center.se.jar
```

The connection parameters window will appear.

1. Enter the login and password to the system (on the first launch use `root:root`, then change the password, see **Staff** on page **287**), and also the private key password.
2. Click **Options** and enter the server connection parameters of the integration module, i.e. its address and port number.
3. Press **External Payment Systems**. The control interface window will open.



Login: root

Password: ····

Name: External payment systems

Address: 10.1.4.22:51010

Keystore Password: ········ ☑ save

Status pane in the lower part of the window displays the current date, time, and connection state. Green spot means that the connection is up, while red indicates that it is down. To repair connection, either click on the status spot or select **System**: **Open connection** in the main menu.

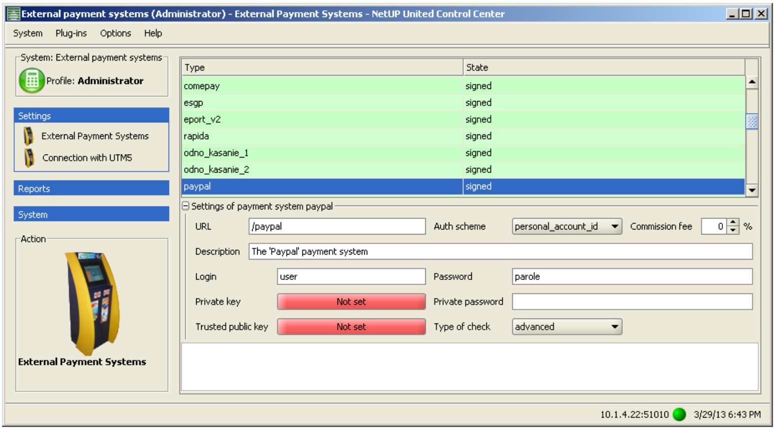
Side effects of turning Internet on and sending a notification, which may or may not be caused by a payment, are controlled by the UTM5 interface parameters (see **Payment systems: Config file** on page **279**) `ext_payment_inet_on` and `ext_payment_notify`, correspondingly.

Control center interface tabs are described below.

Settings

External payment systems

This tab contains the list of all supported payment systems together with their state, which may be either **signed**, **blocked**, or **setting up**.



Type	State
comepay	signed
esgp	signed
eport_v2	signed
rapida	signed
odno_kasanie_1	signed
odno_kasanie_2	signed
paypal	signed

Settings of payment system paypal

URL: /paypal
Auth scheme: personal_account_id
Commission fee: 0 %
Description: The 'Paypal' payment system
Login: user
Password: parole
Private key: Not set
Private password:
Trusted public key: Not set
Type of check: advanced

10.1.4.22:51010 3/29/13 6:43 PM

Bottom pane displays the following parameters for the selected system:

- **URL –** by which the payment system is identified. The payment system integration module awaits requests by the address **https://server:port/url**, where:
  - `server` is the server address;
  - `port` is the port number designated for HTTPS connections, see **Config file** on page **279** (by default 8080);
  - `url` is this parameter’s value.
- **Auth scheme –** is the selector of the parameter to use for authentication. Possible values are:
  - Personal account ID;
  - Login (points to the user’s basic account);
  - IP address (points to the user’s basic account);
  - External ID of the account.
- **Commission fee –** imposed by the payment system.
- **Description –** of the payment system.
- **Login, Password, Private key, Private password, Trusted public key –** are the parameters used for authentication and encryption by some payment systems (see **Settings: Additional parameters** on page **281**).
- **Type of check –** of the payment parameters. Possible values are:
  - **Not specified –** implies the default value (**Advanced**);
  - **Standard –** implies checking of the necessary parameters only;

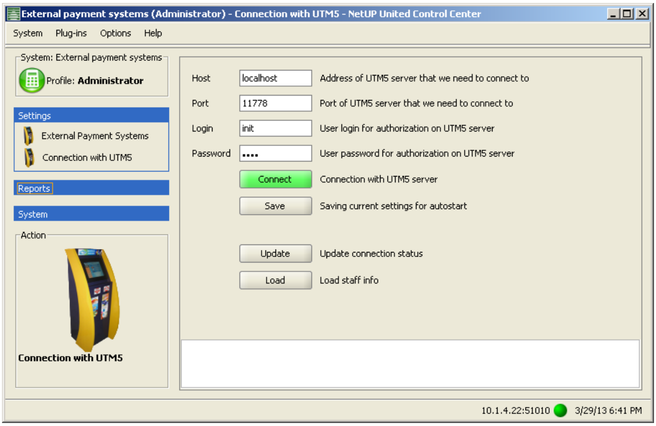
- **Advanced –** implies checking of all parameters;

When using the OSMP payment system, standard checking is compatible with any protocol version, while the advanced checking is compatible only with the older version. The former verifies just that the `txn_date` *parameter contains only digits, while the latter checks it against the template YYYYMMDDHHMMSS.*

- **External –** implies checking by an external utility set in the configs (see **Config file** on page **279**). It is assumed that the utility receives the serialized payment event from the standard input stream, verifies and probably transforms it, and sends the result to the standard output stream. Then the event goes through the standard internal checking routine. If the transformed event cannot be parsed or fails the standard checking, it is discarded.

Connection with UTM5

On this tab you must specify the parameters of connection to the UTM5 core, including server address, port number, login of the system user on whose behalf the payments are to be made, and the corresponding password.



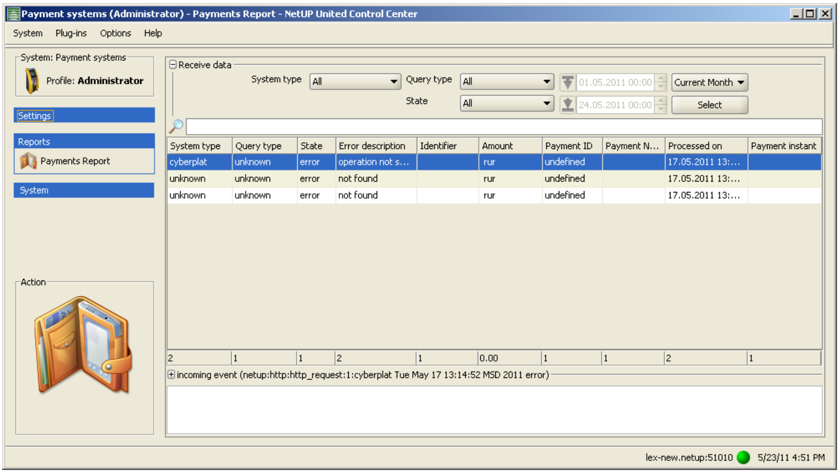
Host	localhost	Address of UTM5 server that we need to connect to
Port	11778	Port of UTM5 server that we need to connect to
Login	init	User login for authorization on UTM5 server
Password	••••	User password for authorization on UTM5 server
	Connect	Connection with UTM5 server
	Save	Saving current settings for autostart
	Update	Update connection status
	Load	Load staff info

10.1.4.22:51010 ● 3/29/13 6:41 PM

The address and port must match those set by the parameters `nxt_v2_bind_host` *and* `nxt_v2_bind_port` *in the main UTM5 config file (see* ***Config file*** *on page* ***279****).*

Payments report

This tab contains the interface for composing reports on payments made via the external payment systems.



System type: All
Query type: All
State: All
01.05.2011 00:00
24.05.2011 00:00

System type	Query type	State	Error description	Identifier	Amount	Payment ID	Payment N...	Processed on	Payment instant
cyberplat	unknown	error	operation not s...		rur	undefined		17.05.2011 13:...	
unknown	unknown	error	not found		rur	undefined		17.05.2011 13:...	
unknown	unknown	error	not found		rur	undefined		17.05.2011 13:...	
2	1	1	2	1	0.00	1	1	2	1

incoming event (netup:http:http_request:1:cyberplat Tue May 17 13:14:52 MSD 2011 error)

lex-new.netup:51010 5/23/11 4:51 PM

The **Receive data** pane contains controls for selecting payments by the following criteria:

- System type (ID of the system which has processed the request).
- Query type (new payment, payment precheck, revoked payment, status, export payments, unknown, or all).
- State:
  - All states;
  - Processed;
  - Error;
  - Success.
- Time of payment.
- Filtering by text search. Filter mode may be set via the context menu as one of the following:
  - All words;
  - Exact phrase;
  - Any word.

After pressing **Select**, a list of matching payments will appear. Each payment is described with the following parameters (note that some columns may be hidden by default):

- **ID –** of the payment in the database.
- **Processed event ID –** in the database.
- **Payment ID –** (external ID of the payment, if applicable).
- **Payment number –** (external number of the payment, if applicable).

- **Identifier –** of the personal account to which the payment is made.
- **Payment instant –** which is the time when the payment has been registered by the system.
- **Processed on –** which is the time when the payment's processing has been finalized.
- **Error description –** (if an error has occurred).
- **Amount –** of the payment.
- **Currency –** of the payment.
- **Query type –** of the payment event.
- **System type –** which is the ID of the payment system that has processed the request.
- **State –** of the payment after processing.

  If a payment is processed normally, its state is set to **Processed**, otherwise it is set to **Error**.

The **Incoming event** pane displays the complete set of parameters of the payment event received from the payment system.

System

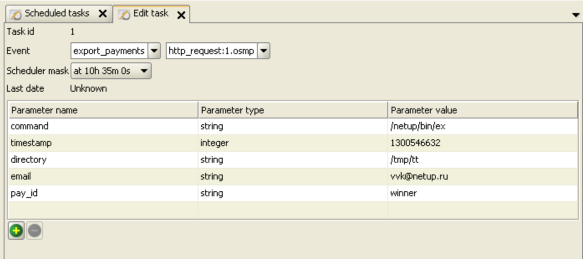
Staff

This tab contains editable list of system accounts. By default there are two of them, namely `root` for working with UCC and `utm` for connecting to UTM5.

Scheduled tasks

This tab contains editable list of all planned tasks in the system. A task consists of an event, a schedule by which it is invoked, and optional parameters passed to it. By pressing **Execute** any event may be initiated at any moment, regardless of the schedule.

To receive the report on payments from OSMP, you must add the corresponding event (see **Events** on page **288**) to the list of tasks:



Task id 1
Event export_payments http_request:1.osmp
Scheduler mask at 10h 35m 0s
Last date Unknown

Parameter name	Parameter type	Parameter value
command	string	/netup/bin/ex
timestamp	integer	1300546632
directory	string	/tmp/tt
email	string	vvk@netup.ru
pay_id	string	winner

Current connections

This tab lists current connections. For each entry the following information is provided:

- **Address** and **Port** where the connection originates;

- **Staff**, i.e. the connecting system user;
- **State** of the connection (authorized / not authorized);
- **Keep alive** flag.

Roles

This tab contains editable list of system roles in a form of a hierarchical tree. Roles assigned to system accounts determine their rights to perform certain system actions.

Events

This tab contains the list of system events. Each event is associated with certain roles, i.e. may be invoked only by the system users sharing any of these roles.

There is a special event related to OSMP system and requesting a report on payments for the given day to be sent to the given e-mail address. This event is called `http_request` (family `export_payments`, version `1.osmp`) and has the following parameters:

- **email –** is a string parameter containing e-mail address to send the report to. Default value is `test@osmp.ru`.
- **pay_id –** is a string parameter containing provider ID in OSMP. Default value is stored in the system settings (see **Config file** on page **279**).
- **timestamp –** is the unix timestamp of the moment to create the report on preceding day. Default value is "now".
- **command –** is a string containing path to the script which actually sends the report. It is called once the report is generated; path to the report file and the target e-mail address are passed to it as command-line parameters. Default value is `/netup/bin/report`.
- **directory –** is the path to location where the report should be placed.

To receive the report every day, create a scheduled task containing the corresponding event (see **Scheduled tasks**).

APPENDIX

Approaches to traffic shaping

Shaping is a limitation of bandwidth for IP traffic customers. The limitation may be of the following types:

- static (constant, defined solely by the tariff plan);
- dynamic (may depend on time and on the amount of traffic consumed).

UTM5 provides interface for setting up both static and dynamic shaping for selected services and tariff plans.

Actual bandwidth regulation occurs on the traffic routers, which may be PC-routers, Cisco routers, etc. The billing software may interact with those in following manners:

1. Using external scripts. On some external event (say, when the traffic exceeds some threshold) the billing system starts an external script controlling the shaper to change the bandwidth or probably break the connection. The user’s IP address and the new bandwidth may be passed to the script as parameters.

   The script calls external traffic control utilities (for example, `tc` from the `iproute2` package for Linux, or `ipfw` for FreeBSD with enabled `dummynet`). Usage of these utilities may require additional tuning of the OS and/or other software.
1. Using RADIUS attributes (for VPN and dialup services). A response given by the RADIUS server to an authentication request may include one or several attributes controlling the connection bandwidth for the given user on the given NAS, if the NAS supports this functionality. Cisco routers are an example of such NAS.

   In this way the bandwidth is set for a connection permanently as it is established, so any corrections will have to wait till the next connection.

   These two methods may be used either simultaneously as well as separately.

UTM5 uses the following approaches to shaping:

1. In case of shaping by the external scripts, the parameters to pass to the script are set up on the **Firewall rules** page (see **UTM5 RFW** on page **185** for more expanded description of RFW workflow and **Administrator’s interface: Firewall rules** on page **86** for interface details). Each rule is associated with one or several events that trigger the script with the prescribed parameters. Path to the script is set via the `firewall_path` variable in the `utm5_rfw.cfg` config file.
   - Static shaping may be done by the rules linked to the **Internet on** event.
   - Dynamic shaping is possible in case if the corresponding module is present, and is set up by the rules linked to the events **Set bandwidth limit**, **Edit bandwidth limit** and **Delete bandwidth limit** on incoming or outgoing channel. Events of the first two kinds occur

when the traffic amount passes over some predefined borders, and the **Delete bandwidth limit** happens at the end of accounting period or when a service link is deleted. The borders can be made time-dependent, so that the switch of time ranges may also fire the events of these types. See **Administrator's interface: Dynamic shaping** on page **96**.

The dynamic shaping module requires a separate license.

1. In case of RADIUS attributes-driven shaping:
    - Static shaping is set up on the **RADIUS Parameters** tab on **Service** window (see **Administrator's interface: Services** on page **61**).
    - Dynamic shaping is possible in case if the corresponding module is present, and is set up on the **RADIUS parameters** tab of the **Dynamic shaping** page under **Settings** (see **Administrator's interface: Dynamic shaping** on page **96**). Certain RADIUS attributes may be provided for each service in order to limit the bandwidth depending on the traffic amount. Dynamic adjustment of attributes is enabled by the use of variables.

RADIUS parameters

Below is an example of RADIUS attribute for dynamic shaping to use with Cisco router.

- **Vendor –** is set to 9;
- **Attribute –** is set to 1;
- **Attribute type –** is set to string;
- **Value –** is set to

```
lcp:interface-config#1=rate-limit input IN_BANDWIDTH_BITS
   IN_CISCO_NORMAL_BURST IN_CISCO_EXTENDED_BURST
   conform-action transmit exceed-action drop
```

Before sending this string to NAS, the system substitutes the `IN_BANDWIDTH_BITS` variable with the numeric value of bandwidth for the given user (in bits/sec), determined from the current time and consumed traffic amount, as prescribed by the shaping settings. The following two variables are interpreted as follows:

`IN_CISCO_NORMAL_BURST` is a number of bytes to send in one burst. It is calculated as the number of bytes passing in 1.5 seconds at given bandwidth.

`IN_CISCO_EXTENDED_BURST` is the possible amount of bytes above burst size to send in one interval. If spent, it must be compensated at the cost of bandwidth at periods when the load is below maximum. Its numeric value is twice larger than the normal burst size.

The rest of instructions state that the traffic within the given bandwidth and the allowed excesses is to be passed through (`conform-action transmit`), while the extra packets after the depletion of excess burst size are ignored (`exceed-action drop`).

External scripts

The examples of executable files are presented below. It is supposed that the firewall rules for **Set** bandwidth limit, **Edit** bandwidth limit, and **Delete** bandwidth limit are already created, and each of them lists the script parameters in the following form:

```
UID UIP UBITS UMASK BANDWIDTH [0|1|2]
```

First five parameters are the variables to be replaced with their values on execution:

`UID`	user ID
`UIP`	user IP address
`UMASK`	user network mask
`UBITS`	binary network mask
`BANDWIDTH`	current connection bandwidth

The last parameter is either 0, 1, or 2, depending on the nature of the event:

- `0` – **Delete** bandwidth limit;
- `1` – **Set** bandwidth limit;
- `2` – **Edit** bandwidth limit.

Linux

Below is the example for the `iproute2` software shaper running on GNU/Linux.

It is assumed that the following commands have been run in advance:

```
tc qdisc add dev eth0 root handle 1: htb
```

– a queueing discipline (`qdisc`) with ID=1 is created on the incoming interface `eth0`. The simple and quick Hierarchy Token Bucket (`htb`) method of ordering is set for the queue.

```
tc class add dev eth0 parent 1: classid 1:1 htb rate 100mbit ceil
100mbit burst 200k
tc class add dev eth0 parent 1:1 classid 1:10 htb rate 1mbit burst
20k
```

– a root class with ID 1:1, guaranteed bandwidth of 50 Mbit/sec (`rate 50mbit`), and peak bandwidth of 100 Mbit/sec when unoccupied bandwidth is available (`ceil 100mbit`), passed through in a 200 KB bursts (`burst 200k`) is created for the queue. It will be used as the parent for all other classes and distribute the bandwidth among them with the ability to borrow the unoccupied portion from each other. Also, a class with ID 1:10 is created with bandwidth 1 Mbit/sec without borrowing, to let through the unidentified traffic from the queue.

```
tc filter add dev eth0 parent 1: protocol ip prio 3 handle 1 fw
classid 1:10
```

– a filter is created to direct the uncategorized traffic from the queue to the class with poor bandwidth.

Later on, when the script is called due to the **Set** bandwidth limit events, it works as follows:

- an iptables rule is created to mark the traffic packets incoming to the given IP address (outgoing traffic may be shaped separately in a similar manner);
- a new filter is created sending the marked traffic into the new class;
- a new class is created with the given bandwidth.

On **Edit** bandwidth limit event the bandwidth of the class is altered, and on **Delete** bandwidth limit the class is deleted together with the corresponding filter and the `iptables` rule.

The traffic to which no filters apply (i.e. belonging to the users for whom the shaping is not set up) skips the queue and passes directly.

The script itself is given below:

```
#!/bin/bash if="eth1" echo $* echo "First create: tc qdisc add
dev $if root handle 1: htb" case "$6" in
0) iptables -t mangle -D FORWARD -s 0/0 -d $2/$3 -j MARK
--set-mark $1 tc filter del dev $if parent 1: protocol ip prio
3 handle $1 fw classid 1:$1 tc class del dev $if parent 1:1
classid 1:$1 htb rate $5kbit burst 20k;;
1) iptables -t mangle -A FORWARD -s 0/0 -d $2/$3 -j MARK
--set-mark $1 tc filter add dev $if parent 1: protocol ip prio
3 handle $1 fw classid 1:$1 tc class add dev $if parent 1:1
classid 1:$1 htb rate $5kbit burst 20k;;
2) tc class change dev $if parent 1:1 classid 1:$1 htb rate
$5kbit burst 20k;; *) echo "Usage: `basename $0` {UID UIP UBITS
UMASK BANDWIDTH [0|1|2]}" >&2 exit 64;; esac
```

FreeBSD

Below is an example script for `dummynet` working on FreeBSD.

When the script is called on **Set** bandwidth limit, a pipe with limited bandwidth is created together with the rule that directs the incoming traffic of certain user on the `em0` interface to this pipe. On **Edit** bandwidth limit pipe bandwidth is changed. On Delete bandwidth limit the pipe and the corresponding rule are deleted.

Script is designed to work in multipass regime (`sysctl net.inet.ip.fw.one_pass` must return 0).

```
#!/bin/sh case "$6" in 0) /sbin/ipfw delete $1 /sbin/ipfw pipe
delete $1 ;; 1) /sbin/ipfw pipe $1 config bw $5Kbit/s
/sbin/ipfw add $1 pipe $1 ip from any to $2/$3 via em0 ;; 2)
/sbin/ipfw pipe $1 config bw $5Kbit/s ;; esac
```

Template variables

This section contains the list of variables which may be used in templates of various types (see **Document templates** on page **93**).

Variables

Template variables are split into several groups:

- **Document**

Name	Type	Description
document.number	int32	Document number
document.date	int32	Document creation date

- **User**

Name	Type	Description
user.id	int32	User ID
user.full_name	string	Full name of the user
user.login	string	User login name
user.password	string	User password
user.actual_address	string	Actual address
user.juridical_address	string	Legal address
user.home_telephone	string	Home phone
user.work_telephone	string	Work phone
user.mobile_telephone	string	Mobile phone
user.tax_number	string	ITIN
user.kpp_number	string	Reg. code
user.icq_number	string	ICQ number
user.web_page	string	Web page
user.district	string	District
user.building	string	Building
user.entrance	string	Entrance
user.floor	string	Floor
user.flat_number	string	Flat number
user.personal_manager	string	Personal manager
user.basic_account	int32	Basic account ID
user.passport	string	Passport
user.email	string	E-mail
user.comments	string	Comments
user.bank_account	string	Bank account
user.bank_name	string	Bank name
user.bank_city	string	Bank city
user.bank_bic	string	BIN
user.bank_corr_account	string	Bank corr. account number
user.currency_short_name	string	Currency short name
user.currency_full_name	string	Currency full name

Name	Type	Description
user.currency_code	int32	Currency code
user.params.{param_id} [a]	string	Additional user parameter with ID {param_id}
user.contacts.{contact_id}.email [b]	string	Additional contact e-mail with ID {contact_id}
user.contacts.{contact_id}.full_name	string	Additional contact full name with ID {contact_id}
user.contacts.{contact_id}.short_name	string	Additional contact short name with ID {contact_id}
user.contacts.{contact_id}.position	string	Additional contact position with ID {contact_id}
user.contacts.{contact_id}.reason	string	Additional contact description with ID {contact_id}
user.contacts.{contact_id}.telephone	string	Additional contact phone number with ID {contact_id}

a. {param_id} is an integer value of the ID of an auxiliary user parameter

b. {contact_id} may take the following values: headman, booker (accountant) or an integer additional contact ID, starting with one

- **Personal account**

Name	Type	Description
account.account_id	int32	Account ID
account.external_id	string	External account ID
account.balance	double	Balance
account.credit	double	Credit
account.vat_rate	double	VAT rate
account.sale_tax_rate	double	Tax rate
account.access_card_number	string	IPTV access card number

- **Provider**

Name	Type	Description
provider.full_name	string	Provider name

Name	Type	Description
provider.short_name	string	Provider short name
provider.juridical_address	string	Legal address
provider.actual_address	string	Actual address
provider.tax_number	string	ITIN
provider.kpp_number	string	Reg. code
provider.chief_full_name	string	CEO
provider.chief_short_name	string	CEO: short name
provider.booker_full_name	string	Accountant name
provider.booker_short_name	string	Accountant: short name
provider.bank_account	string	Bank account
provider.bank_name	string	Bank name
provider.bank_city	string	Bank city
provider.bank_bic	string	BIN
provider.bank_corr_account	string	Corr. account number

- **Contract**

Name	Type	Description
contract.number	int32	First contract ID
contract.name	string	First contract name
contract.date	int32	First contract creation date
contract.{contract_id}.number (a)	int32	ID of a contract # {contract_id}
contract.{contract_id}.name	string	Contract # {contract_id} name
contract.{contract_id}.date	int32	Contract # {contract_id} date

a. {contract_id} is the user’s contract ID, starting with one

- **Payment**

Name	Type	Description
payment.id	int32	Payment transaction ID
payment.amount_in_currency	double	Amount in used currency

Name	Type	Description
payment.amount_absolute	double	Amount in system currency
payment.date.actual	int32	Actual payment date (when it is registered by UTM5)
payment.date.enter	int32	Payment date (when it was made by user)
payment.date.burn	int32	Payment expire date
payment.document_number	string	Payment document number
payment.comments.user	string	Comments for user
payment.comments.admin	string	Comments for administrator
payment.hash	string	Payment hash
payment.currency_rate	double	Currency rate
payment.currency_short_name	string	Currency: short name
payment.currency_full_name	string	Currency: full name
payment.currency_code	int32	Currency code

- **Bill**

Name	Type	Description
bill.sum_without_tax	double	Sum w/o taxes
bill.sum_with_tax	double	Sum, including taxes
bill.size	int32	Number of lines in bill
bill.period_start	int32	Period start date
bill.period_end	int32	Period end date
bill.balance_when_created	double	Balance at the moment when bill was created
bill.debt	double	Debt
bill.payment_amount	double	Payment amount w/o taxes
bill.payment_amount_with_tax	double	Payment amount with taxes
bill.date	int32	Date

- **Call details**

Name	Type	Description
summary.periodic_fee	double	Periodic service fees
summary.total_fee	double	Telephony service fees
summary.other_fee	double	Other services fees
summary.local.charges	double	Local call charges
summary.local.count	double	Local calls number
summary.local.duration	double	Local calls duration
summary.innerzone.charges	double	Same for inner zone calls
summary.innerzone.count	double	
summary.innerzone.duration	double	
summary.intercity.charges	double	Same for intercity calls
summary.intercity.count	double	
summary.intercity.duration	double	
summary.international.charges	double	Same for international calls
summary.international.count	double	
summary.international.duration	double	

Iterating variables

This section contains the list of iterating variables. When generating a document, these variables are replaced with an array of values. They should be placed in a table row in a document template. In this case the number of rows in the table is automatically increased to hold all the values that are returned.

Iterating variables also are split into several groups:

- **IP group table iterators**

Name	Type	Description
ipgroup.login	string	IP group login
ipgroup.password	string	IP group name
ipgroup.mac	string	MAC address
ipgroup.ip	string	IP address

Name	Type	Description
ipgroup.mask	string	Subnet mask
ipgroup.gateway	string	Gateway

IP group table iterators only include non dynamically created IP groups that have a non empty Login field and a non zero IP address

- **Connected tariff plans table iterators**

Name	Type	Description
tariff.name	string	Plan name
tariff.cost	double	Cost
tariff.account_id	int32	Personal account ID

- **Bill iterators**

Name	Type	Description
bill_entry.id	int32	Entry index, starting with one
bill_entry.name	string	Entry name
bill_entry.price	double	Entry price
bill_entry.quantity	double	Quantity
bill_entry.sum_with_tax	double	Sum, including taxes
bill_entry.sum_without_tax	double	Sum w/o taxes
bill_entry.tax	double	Tax
bill_entry.tax_rate	double	Tax rate
bill_entry.unit_name	string	Unit name (returns replacement key)
bill_entry.unit_code	string	Unit code (returns replacement key)
bill_entry.alt.price	double	Alternative price
bill_entry.alt.quantity	double	Alternative amount
bill_entry.alt.unit_name	string	Alternative unit name (returns replacement key)
bill_entry.alt.unit_code	string	Alternative unit code (returns replacement key)

- **Call details iterators**

Name	Type	Description
call.id	int32	Call ID
call.zone	string	Zone name
call.direction	string	Direction name
call.date	int32	Call date
call.calling_number	string	Calling number
call.called_number	string	Called number
call.called_prefix	string	Called prefix
call.duration	int32	Call duration
call.type	string	Call type (returns replacement key)
call.cost	double	Call cost

- **Iterators of service link parameters for dialup service**

Name	Type	Description
dialup.login	string	Login name
dialup.password	string	Password
dialup.cid	string	CID parameter value
dialup.csid	string	CSID parameter value

- **Iterators of service link parameters for hotspot service**

Name	Type	Description
hotspot.login	string	Login name
hotspot.password	string	Password

- **Iterators of service link parameters for telephony service**

Name	Type	Description
telephony.login	string	Login name
telephony.password	string	Password
telephony.number	string	Telephone number
telephony.incoming_trunk	string	Incoming trunk
telephony.outgoing_trunk	string	Outgoing trunk
telephony.pbx	string	PBX ID parameter value

Name	Type	Description
telephony.cid	string	CID parameter value

Template types

Depending on the template type, it may include variables from the following groups:

	Invoice	Commercial invoice	User memo	Certificate	Receipt	Contract	Call details
Document	•	•	•	•	•	•	•
User	•	•	•	•	•	•	•
Personal account	•	•	•	•	•	•	•
Provider	•	•	•	•	•	•	•
Contract	•	•	•	•	•	•	•
Payment					•		
Bill	•	•		•			
Call details							•
Bill iterators	•	•		•			
IP group table iterators			•			•	
Connected tariff plan table iterators			•			•	
Call details iterators							•

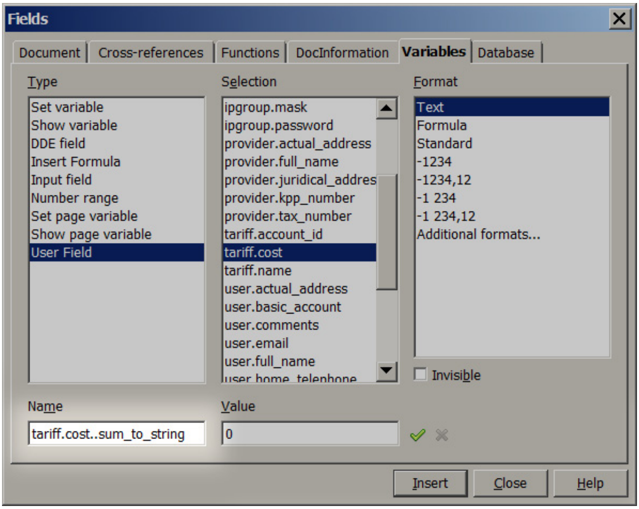
Variable modifiers

Variable modifiers modify the values returned for variables. The following modifier types are available:

Name	Argument type	Result type	Description
translate	string	string	Replaces with a value from the replacements list if the variable and the key match [a]
replace	string	string	Replaces matching part of the variable with the value from the replacements list
date_short	int32	string	Date format DD/MM/YYYY
date_long	int32	string	Date format “DD” Month YYYY
date_time	int32	string	Time format MM.DD HH:MM
duration	int32	string	Duration format HH:MM:SS
sum_to_string	double	string	Sum to string

a. See Administrator’s interface: Replacements in documents on page 85

In order to use a variable modifier, add its name after the name of a variable, separated by two dots:



Name: tariff.cost..sum_to_string

Value: 0

Then insert the modified variable into the template.

Phone:	+7 495 510 1025
Fax:	+7 499 783 0080
Address:	Russia, Moscow, Ulofa Palme str. 1, sect. 7
Post:	119311, Russia, Moscow, P.O. Box #87
E-mail:	info@utm-billing.com
Web:	http://www.utm-billing.com