

Alaris MNP Server

User's Guide

version 2.5.0

Document Type: User's Guide

Date of Issue: 7/2/2026

Copyright © 2005 - 2026 Alarislabs Pte Ltd.
All rights reserved.

The information contained in this document is the property of Alarislabs Pte Ltd. No part of this publication may be reproduced or copied in any form or by any means - graphic, electronic or mechanical including photocopying, recording, taping, or any other information storage and retrieval System - without written consent of Alarislabs Pte Ltd. No third party, organization or individual, is authorized to grant such permission.

The information provided in this document is for general informational purposes only. While we strive to ensure accuracy and completeness, the documentation may contain omissions, or may not reflect the most current updates or include all recent changes. Alarislabs Pte Ltd. cannot be held responsible for any inaccuracies or incomplete information contained in this manual. Users are encouraged to verify details and consult with the Alaris technical support team for the most up-to-date and comprehensive information (support@alarislabs.com). By using this manual, you acknowledge that you understand and accept these terms.

Table of Contents

1 MNP server overview	4
2 Account settings	5
3 Sources	7
4 Clients	16
5 History logs	22
6 Reselling API	24
6.1 HTTP	24
6.2 ENUM	25
6.3 Camara	26
7 Dipping rules	29
7.1 How to configure LCR-based HLR routing	32
8 Camara API Rules	34
9 Settings	36
9.1 Settings	36
9.2 Currencies	37
9.3 Exchange rates	38
10 Import rates	40
11 Direct dipping	42
12 Simulation	45
13 Response code translations	48
13.1 System codes	48
13.2 Translation rules	49
14 Users	52
15 Roles	54
15.1 Permissions	55

MNP server overview

1 MNP server overview

MNP server is an add-on module of Alaris SMS Platform that serves to manage number portability and HLR reselling. It serves to configure supported HLR providers (sources), Dipping rules and HLR clients. To enable the module and make it available in the Alaris SMS Platform interface for HLR reselling purposes, contact your Account manager. For regular configuration of Dipping rules, submit a ticket on our HelpDesk.

To access the MNP server, proceed as follows:

1. Access the MNP server using a separate URL provided by the Alaris technical support team or the one registered by the System owner. For users that purchased the HLR reselling feature at Alaris SMS Platform, the MNP server can be accessed through the Alaris SMS Platform interface.

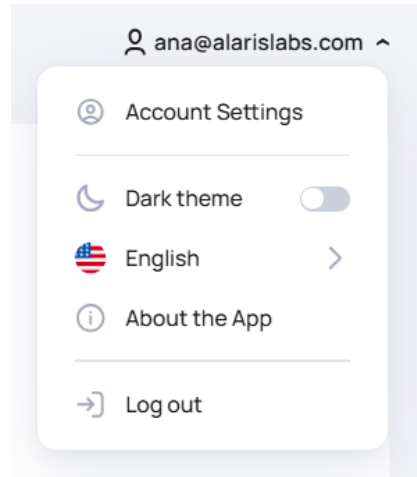
NOTE: For purchasing, contact your Account manager.

2. The interface contains the following pages: [Sources](#)^[7], [Clients](#)^[16], [Reselling API](#)^[24], [Dipping rules](#)^[29], [Settings](#)^[36], [Import rates](#)^[40], [Direct dipping](#)^[42], [Response code translations](#)^[48], [Users](#)^[52] and [Roles](#)^[54]. Each page (except [Settings](#)^[36]) contains a table of records and the following buttons at the top:
 - a. *Refresh page*
 - b. *Export*: serves to export the table into an MS Excel file.
 - c. *Add new* (user, rule etc.): serves to add a new record.
3. Configure the settings as detailed below.

Account settings

2 Account settings

The *Account settings* menu is available if the user has logged in using a separate URL provided by the Alaris technical support team. When accessed through Alaris SMS Platform interface, it is not displayed.



Account settings

Use the *Dark theme* toggle to apply the dark theme. Select the interface language with the help of language switcher. Click *About the App* to view the installed version of the product and the time zone. Log out using the *Log out* button.

Click *Account settings* to be forwarded to *Your profile* page where personal information can be filled in.

A screenshot of a 'Your profile' page. At the top left is the title 'Your profile' and at the top right is a user profile icon followed by 'kate@example.com' and a downward arrow. The main content area is a white card with a rounded shadow. It has a section titled 'Personal information' with four input fields: 'First name' (containing 'Kate'), 'Last name' (empty), 'Email' (containing 'kate@example.com'), and 'Login' (containing 'kate'). Below these is a 'Page size' dropdown menu set to '25'. There is a 'Password' section with a note 'We recommended that you change your password periodically' and a 'Change password' button. At the bottom of the card are two buttons: 'Save changes' (blue) and 'Cancel' (white).

Your profile

Account settings

Select a value from the *Page size* list to define the number of records within one page on [Sources](#)^[7], [Clients](#)^[16], [Dipping rules](#)^[29] shown by default.

Use the *Change password* button to set a new password.

Sources

3 Sources

The page allows adding a new HLR service that can be used in [Dipping rules](#) ²⁹ as an HLR source. The main table shows the existing sources that can be filtered with the help of the *Name* and *Type* filters.

Sources

iana@alarislabs.com

Total: 6 Filter Refresh page Export Add new source

NAME source	×	TYPE	CONNECTION TYPE	HUNTSTOP	TTL
AP Source 1		mitto	HTTP	Disabled	50
AP Source 2		ccs	HTTP	Disabled	50
AP Source 3		akton	HTTP	Disabled	50
CNsource		akton	HTTP	Disabled	0
SourceForRename		akton	HTTP	Disabled	0
camara_source		camara_kyc_match	CAMARA		0

Sources

Sources



Add new source



ENUM

HTTP

SS7

MNP

Camara

Name
mnpType
mnp

General

Cached response lifetime *
0☒ Reroute and keep MCCMNC

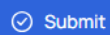
Cache all

Only successful

Only specific

Specify response code list to cache
["1","SUCCESS"]Currency name *
Euro

MNP

MNP stage DB
4MNP active DB
6

Submit

Close

Add new source (part 1 of 2)



Sources

MNP

MNP stage DB
4

MNP active DB
6

Custom MNP: TRASH, UNALLOCATED, RCS_CAPABLE, WA_CAPABLE
RCS_CAPABLE

Huntstop & timeout

Huntstop
Disabled

MCCMNC Translation

MCCMNC translation
{~238001~:~238002~;~238003~:~999999~}

Reset

Submit

Close

Add new source (part 2 of 2)

To add a new service, click the *Add new source* button and configure the following fields (fields marked with an asterisk * are required). Note that for some fields an example can be found by hovering over the field.

- Select the HLR service protocol in the top panel: *ENUM, HTTP, SS7, MNP*.
- Select the HLR service protocol in the top panel: *ENUM, HTTP, SS7, MNP, Camara*.
- *Name*: insert a unique provider name. The name will be used to configure Dipping rules. It is possible to add several sources with different names but of the same type. It may come in handy for configuration of the same HLR provider with different settings. For example, sources under *mitto_1* and *mitto_2* of the *mitto* type can be added with separate time-to-live values. Such a configuration allows separating sources if for some countries TTL must differ
- *Type*: select the provider type from the drop-down list. Basically, it is the list of supported HLR providers. The type is an internal hardcoded name in accordance with which a request to the provider is formed and sent as well as the provider's response is recognized
- *Currency name*: the currency name in which HLR provider's rates will be imported via [Import rates](#)^[40]. The currency is applicable for LCR routing (configured via [Dipping rules](#)^[29])

Based on the selected *Type*, the following additional fields appear.

- *Cache all, Only successful, Only specific*: select the appropriate value from the bar:
 - *Cache all*: all responses will be cached (selected by default).
 - *Only successful*: when selected, the results will be placed to cache only if the HLR MCCMNC was successfully obtained.

Sources

- *Only specific*: specify the responses that must be cached in the field that appears below the bar. The suggested syntax is ["1", "SUCCESS"] - that is, comma-separated values in parentheses, enclosed in square brackets.
- *Do not use MCCMNC* (available if *Type=hlrlookups*): when enabled, only the *hlrResponseCode* value is taken from the vendor's response.
- *Custom MNP* (available if *Type=mnnp*): allows the user to select one of the fields at the MNP source level and return the number details received from the MNP to the client in response. Possible values are: *TRASH*, *UNALLOCATED*, *RCS_CAPABLE*, *WA_CAPABLE*. To return the data, select the appropriate fields (*trash*, *unallocated*, *is_rcs_capable*, *is_wa_capable*) in the *Response fields* list at the client level (for the HTTP client). For ENUM clients, contact the Alaris technical support team and provide the code MNP-516 to configure the ENUM response template.

NOTE: In order to use all fields, create separate MNP sources and specify the required sources in the HLR rule sequentially (*Choice list* field in [Dipping rules](#)^[29]). Custom fields received from such sources will not be overwritten when requesting the next source. Additionally, new data must be imported into MNP databases in the following format (from a csv file): *prefix;boolean value*

for example:

34123456>true

- *MNP active DB* (available if *Type=mnnp*): the number of the active in-memory DB to which the MNP base will be loaded.
- *MNP stage DB* (available if *Type=mnnp*): the number of the inactive in-memory DB.

NOTE: If a single MNP database is used, it is recommended to leave the values empty. It should be noted that the MNP database loading pre-configuration is performed by the Alaris team.

- *Cached response lifetime*: TTL, in seconds. The parameter defines the period to store the HLR results in cache for a specific number. For example, the parameter is set to 10 (seconds). A new request to number 3411111111 is received. Once HLR results are obtained from the HLR service, they are placed in cache. If within the following 10 seconds a request to the same number comes, no actual dipping will be performed but the previous results will be used. The minimum value is 0 (that is, no request will be cached). The maximum value is to be clarified with the Alaris technical support team to avoid memory overflow. Generally it is recommended to specify the value not greater than 604800 (a week)
- *Huntstop*: a huntstop option. Even if the HLR source returns an unsuccessful response (for example, with no HLR MCCMNC in it), no further HLR source will be queried. Possible values are:
 - *Disabled*: huntstop is disabled (default value).
 - *On failed response*: querying of subsequent next-in-line HLR sources stops when an unsuccessful response is received from the HLR source (result=-1,-3). This value is not applicable if the *source type=mnnp*.
 - *On source timeout*: querying of next-in-line HLR sources stops when the response from the source times out (result=-6). This value is not applicable if the *source type=mnnp*.
 - *Unconditional*: previous logic with huntstop enabled (HLR source search stops when result=-1, -3, -6).

NOTE: The response timeout from the ENUM provider is calculated as the product of the parameters *ENUM request timeout* and *Number of request attempts*.

Sources

- *Reroute and keep MCCMNC*: when enabled, the next-in-line HLR source (if any) after the HLR source in question will be requested for the number additionally in order to obtain the HLR response code (*hlrResponseCode*). The HLR MCCMNC and the portability flag will be fetched from the HLR source with the enabled *Reroute and keep MCCMNC* flag that was last in the list. The option comes in handy when the first source is MNP, and provides the info about MCCMNC and portability flag, while the second source is regular HLR and supplies the HLR response code.

NOTE: By default, only the last HLR source will be mentioned in the EDR, therefore the HLR billing happens for the last HLR source only. To bill all HLR sources, an internal parameter must be disabled. For this, contact the Alaris technical support team and communicate the code CS-479.

- *Provider response code list to cache*: the list of *hlrResponseCode* (as an array) for the HLR responses to be cached (applicable if *Only specific* is selected). Example: ["1", "SUCCESS"]
- *Host address for local address*: specify the IP address available for the server scheme which will be used as the source IP address when sending queries if an IP address other than the MNP configurator IP address must be used
- *Status request delay* (available if *Type=unibell*, or *redsms*, or *redsms2*): serves to specify forced delay between requests (if not specified, the value of 2 seconds is used)

HTTP-specific settings:

- *Source URL*: the provider's URL. For example: <http://127.0.0.1:5555/websmpp/hlr/lookup>.
In some cases (when the provider operates following a 2-step request algorithm) it is required to fill in the *URL for first request to provider* and *Status URL* parameters. In addition to the *Source URL* link some providers require the *MNP service domain* (domain) parameter to be passed in a request.
- *Username/Password/API key/User email/Customer ID/Customer key (username)/Secret key (password)*: specify the credentials granted by the HLR provider.
- *HTTP request timeout*: the period in seconds to expect the HLR service response. Once reached, the next-in-line HLR service will be used (if any). If a value is not specified, the value of the Default request timeout setting will be applied ([Settings](#)^[36]).
- *Source GT*: global title (applicable to HLR over SS7).
- *MCC list to handle IMSI from response*: the list of MCCs for which first 6 digits are fetched from the *imsi* field (applicable to HLR over SS7). Example: [262,302,352,354,467,503,751]. For the MCCs that are not in the list, first 5 digits will be parsed as MCCMNCs.
- *Ignore Wireline* (available if the *Type=xconnect*): when enabled, and if *nt=wireline*, the *HLR MCCMNC* will not be applied even if the HLR provider returned it.
- *Ignore mccmnc if ndpi false* (available if the *Type=xconnect*): when enabled, and if *ndpi=false*, the *HLR MCCMNC* will not be applied even if the HLR provider returned it.
- *Number of request attempts, Status request delay* (in milliseconds) (available if the *Type* field contains an HLR source working with two-step requests, for example, *bsg*): serve to configure request retries.

ENUM-specific settings:

- *IP address*: the IP address of the ENUM HLR provider. The valid format is an IP address (no protocol specified) or domain name that starts with *enum* (for example, *enum.eu.provider.com*).

Sources

- *Port*: the port of the ENUM HLR provider.
- *Domain for NAPTR record*: the domain name of the ENUM HLR provider. Example: *e164.arpa*.
- *ENUM request timeout*: the timeout (in ms) to wait for a response from the DNS server. If not set, the default value is 2000 ms.
- *Number of request attempts*: the number of retries to reach the provider's server. Must be greater than 0. If not set, the default value is 3.

NOTE: If the request was resubmitted to the same provider by the parameters *Number of request attempts* and *ENUM request timeout*, interim HLR EDRs will be recorded.

- *Port for local address*: specify a source port from which requests will be sent out. It is recommended to leave the setting empty.

CAMARA-specific settings:

- *Type*: possible values are:
 - *camara_kyc_match*: know your customer (data match), verifies subscriber data against reference information.
 - *camara_num_ver_device_phone_number*: retrieves the phone number for which a *camara_num_ver_verify* request was received, based on the correlator ID sent in the *camara_num_ver_verify* request.
 - *camara_num_ver_verify*: verifies the phone number provided in the request.
 - *camara_sim_swap_check*: verifies whether the subscriber's SIM card was changed within a specified period.
 - *camara_sim_swap_retrieve_date*: retrieves the date of the last SIM card change.
- *Source URL*: URL for sending requests to the service.
- *Credentials*: specify the credentials granted by the Camara provider.
 - *Username, Password*: credentials to generate OAuth token (serve as Client ID and Client secret)
 - *OAUTH Token URL*: URL for token retrieval.

Translation-related settings:

- *Use fixed MCCMNC*: when the checkbox is selected and the HLR provider returns *type=fixed*, a fixed MCCMNC will be assigned (for example, for the *tmthttp* HLR service, it will be 999001; for the *netnumbercid* source, it will be 999999).
- *Use error code from error (E) field* (applicable to *tmtenum*): select to obtain values for the *hlrResponseCode* routing metric from the *e* field of the provider's response. When deselected, the values are fetched from the *gsmcode* field or the *pres* field if the *gsmcode* field is not available.
- *Translate OCN to MCCMNC*: set translation rules to assign the MCCMNC as a pre-set one in accordance with the *ocn* value from the provider's response. The format must be JSON. Example: `{"8304":"302002", "8202":"302003"}` where for *ocn=8304* MCCMNC 302002 will be assigned.

Sources

- *Use error code from Status field:* select to obtain values for the *hlrResponseCode* routing metric from the *status* dictionary of the provider's response. When deselected, the values are fetched from the *error* dictionary.
- *Invalid number code:* specify a value (for example, 99) if it must be used as the *hlrResponseCode* routing metric provided that the HLR service returns *status_message=Invalid Number*.
- *Response code source:* for *netnumber_mnis*: specify *nnti* or *nnci* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Otherwise the values are parsed from the *status* field. For *mitto*: specify *absent* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Specify *complex* to enable the logic as follows: if *absent* is not available in the HLR source response, no *hlrResponseCode* is assigned, if *absent* is returned as true, *hlrResponseCode* is set to 1, if *absent* is returned as false, *hlrResponseCode* will be fetched from *gsmCode* (or, if not available, from *response* with "r" prefix). Otherwise the values are parsed from the *gsmCode/response* fields.

For *tmolive*: specify *error* or *present* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Otherwise the values are parsed from the *status* field.

For *xconnect* and *netnumbercid* types: specify a field from the provider's response (for example, *cic*) to fetch its values to *hlrResponseCode*.

For *datafon_enum* provider: specify *gsm_error_code* in the *Response code source* field of the HLR source. Otherwise the values are parsed from the *err* field.

For *tyntec*: specify *presence* in the *Response code source* field of the HLR source. Otherwise the values are parsed from the *errorCode* field.

- *MCCMNC translation map* and *MCCMNC translation map extra*: set mapping rules to assign the MCCMNC in accordance with the provider's response. The format must be JSON. Example: `{"724301":18, "724302": 54}`

where 724301 and 724302 are the resulting MCCMNCs and 18 and 54 are codes from the provider's response. To double-check from which fields the codes are collected, consult the Alaris technical support team.

- *Use fixed MCCMNC:* when the checkbox is selected and the HLR provider returns *type=fixed*, a fixed MCCMNC will be assigned (for example, for the *tmhttp* HLR service, it will be 999001; for the *netnumbercid* source, it will be 999999).
- *Use error code from error (E) field:* select to obtain values for the *hlrResponseCode* routing metric from the *e* field of the provider's response. When deselected, the values are fetched from the *gsmcode* field or the *pres* field if the *gsmcode* field is not available.
- *Translate OCN to MCCMNC:* set translation rules to assign the MCCMNC as a pre-set one in accordance with the *ocn* value from the provider's response. The format must be JSON. Example: `{"8304":"302002", "8202":"302003"}` where for *ocn=8304* MCCMNC 302002 will be assigned.
- *Use error code from Status field:* select to obtain values for the *hlrResponseCode* routing metric from the *status* dictionary of the provider's response. When deselected, the values are fetched from the *error* dictionary.
- *Invalid number code:* specify a value (for example, 99) if it must be used as the *hlrResponseCode* routing metric provided that the HLR service returns *status_message=Invalid Number*.

Sources

- *Response code source*:
- For *netnumber_mnis*: specify *nnti* or *nnci* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Otherwise the values are parsed from the *status* field.
- For *mitto*: specify *absent* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Specify *complex* to enable the logic as follows: if *absent* is not available in the HLR source response, no *hlrResponseCode* is assigned, if *absent* is returned as true, *hlrResponseCode* is set to 1, if *absent* is returned as false, *hlrResponseCode* will be fetched from *gsmCode* (or, if not available, from *response* with "r" prefix). Otherwise the values are parsed from the *gsmCode/response* fields.
- For *tmltlive*: specify *error* or *present* to gather values for the *hlrResponseCode* routing metric from the corresponding field of the provider's response. Otherwise the values are parsed from the *status* field.
- For *xconnect* and *netnumbercid* types: specify a field from the provider's response (for example, *cic*) to fetch its values to *hlrResponseCode*.
- For *datafon_enum* provider: specify *gsm_error_code* in the *Response code source* field of the HLR source. Otherwise the values are parsed from the *err* field.
- For *tyntec*: specify *presence* in the *Response code source* field of the HLR source. Otherwise the values are parsed from the *errorCode* field.
- For *unibell_v2*: the *Response code source* field can be set to the name of a field from the HLR source response. If the field is left empty, *hlrResponseCode* will be taken from the *status* field.
- *MCCMNC translation map* and *MCCMNC translation map extra*: set mapping rules to assign the MCCMNC in accordance with the provider's response. The format must be JSON. Example: `{"724301":18, "724302": 54}`
- *MCCMNC translation*: set translation rules to convert the MCCMNC from the provider's response to a pre-set one. The format must be JSON. Example: `{"238001":"238002", "238003": "999999"}`.
- *Translate error code to MCCMNC*: set rules to assign the MCCMNC in accordance with the codes returned by the providers. The format must be JSON. Example: `{"0":"238002", "r1": "999999"}`
 - *Extra fields in response* (available if *Type=redsms*): when enabled, the *fields* field with the value *all* will be sent to the provider in the request. In this case, the provider's response contains the *mccmnc* field, from which the data is taken to define the HLR MCCMNC.
 - *Forced error code translation*: enable to assign the MCCMNC in accordance with the *Translate error code to MCCMNC* option even if the HLR MCCMNC was returned by the HLR service.
 - *Mapping carrier_id from response to MCCMNC*: specify a JSON array to assign an MCCMNC in accordance with the *carrier_id* value from the provider's response. Example: `{ "1" : "440050", "2" : "440020", "3" : "440010", "6" : "440011" }`.
 - *Custom translation* (applicable to *restcom*): optional field, when filled in, the *MCCMNC translation* is not verified. Specify a JSON array to assign an MCCMNC in accordance

Sources

with the *imsi* value (symbols from 3 to 5 of the field) from the provider's response.
Example: {"123": "238002", "231": "238003"}.

- *Translate network name to MCCMNC*: specify a JSON array to assign an MCCMNC in accordance with the network name from the provider's response. Example: {"Algar Telecom" : "724032", "Claro" : "724005"}.
- *Inactive subscriber MCCMNC*: specify the MCCMNC which will be assigned as the HLR MCCMNC if *presence=false* in the provider's response. If left empty, no special translation takes place.
- *Translate MNP code to MCCMNC*: set MCCMNC in accordance with the *mnpcode* field of the provider's response. The logic of assignment is as follows: the third, the fourth and the fifth digit of *mnpcode* is used to define the MCCMNC from the translation set.

Clients

4 Clients

The *Clients* page serves to create a new HLR reselling client or manage the existing ones. The main table shows the available clients. To filter the table by the client name, click  *Filter*, unwrap the *Client product* list and either insert the product name, or select the product(s) from the list.

 **Clients**

ana@alarislabs.com 

Total: 8	 Filter	 Refresh page	 Export	 Add new client
NAME	CLIENT PRODUCT	INTERFACES	MAX SESSIONS	
alaris	KA_HLR - adv2 (28746)	enum smsrt http	1000	
ka	TK_CP_client - TK-HRL-edit (29009)	http	1000	
Kate	KA_cli - EUR_HLR (28661)	http	1000	

Clients

To adjust the parameters of an existing client, click the *Edit*  button. Use the *Delete*  button to remove the record. A warning message will appear.

Clients



Add new client



Name

Alice

Login

admin

Password

.....



Generate

Enable OAuth ☒

OAuth Scopes *

Sim swap (check) ×

Sim swap (retrieve date) ×



Number verification (verify) ×

Number verification (phone number) ×

Client product

Alexander YouTube - HLR (EUR) (24589)



Interfaces *

http ×

smsrt ×

☒ Allow debug☐ Remove leading zero from MNC☐ Force result to 1

Response fields *

DNIS ×



Limitations

☒ Submit

Close

Add new client (part 1 of 2)



Clients

Limitations

Max sessions

1000

☒

Request limit

Period type

hour

▼

Period number

5

25.06.2026, 16:30:24

Reset

Submit

Close

Add new client (part 2 of 2)

To add a new record, click the *Add new client* button. The mandatory parameters are as follows:

- *Name*: the client name that can be specified when working with [Camara API Rules](#)
- *Login*: the client login that will be used for authentication if requests are to be sent over HTTP
- *Password*: the client password that will be used for authentication if requests are to be sent over HTTP. Apply the *Generate* button to generate the password
- *Enable OAuth*: enables/disables OAuth authorization. When enabled, the values of the *Login* and *Password* fields are used as the Client ID and Client Secret for token generation. The mandatory *OAuth Scopes* drop-down list also becomes available, containing the following service values accessible to the client:
 - *Sim swap (check)* — verifies whether the subscriber's SIM card was changed within a specified period.
 - *Sim swap (retrieve date)* — retrieves the date of the last SIM card change.
 - *Number verification (verify)* — verifies the phone number provided in the request.
 - *Number verification (phone number)* — retrieves the phone number for which a *Number verification (verify)* request was received, based on the correlator ID sent in the same step.
 - *Know your customer (data match)* — verifies subscriber data against reference information.

Multiple services may be selected for a single client. For the functionality to work correctly, the *http* value must be selected in the *Interfaces* list for the client. In addition, reselling must be enabled (to be agreed separately with an AlarisLabs manager).

- *Client product*: select the HLR client product that has been created in the System. The requests received under the client record will be associated with the client product in order to use its rates

Clients

and issue invoices. The parameter is optional and can be set to *None* if no client product must be assigned to client's requests. Use the *Filter* window to quick-filter and find the required product (the product ID from Alaris SMS Platform is displayed next to the product name).

- *Interfaces*: select the protocols over which the client will send requests. The default value is HTTP. The possible values include *HTTP*, *ENUM*, *SMSRT* (the System value which is used for System configuration)
 - *Remove leading zero from MNC*: select the flag if 5-digit MCCMNCs must be returned to the client (for example, 23801 instead of 238001)
 - *Force result to 1*: if enabled, the result field in the response to the client will be forced to 1 when the number is recognized as ported (ported=1).
 - *ENUM address list* (available if the *enum* value is selected in the *Interfaces* field): list of ENUM IP addresses in the format: *IP address/mask*.
 - *Use Login to authorize client* (available if the *enum* value is selected in the *Interfaces* field): when enabled, the sent *Login* will be verified along with the ENUM IP addresses.
 - *Response fields*: the field defines the list of fields to be returned to the client in response over HTTP. The default value includes:
 - *mccmnc*: the HLR MCCMNC (6-digit by default)
 - *result*: shows if the result is successful (0: yes, other values: no)
 - *ported*: the portability information (0: not ported, 1: ported)
 - *source_name*: the HLR service name (in case of several HLR services, the name of the last HLR service)
 - *source_type*: the HLR service type (in case of several HLR services, the type of the last HLR service)
 - *dnis*: the destination address in e.164 format for which a query was processed
 - *login*: the client login
 - *system_response_code*: System response code values. Possible returned values (and general logic of their assigning) are as follows:
 - 0 - the response code from the HLR provider is successful (the number is available in the network, it is valid etc.) and, if the provider returns an MCCMNC, the HLR MCCMNC is received
 - 1 - the response code from the HLR provider is unsuccessful (for example, the number is unavailable or invalid), but an HLR MCCMNC was received
 - 2 - the response code from the HLR provider is unsuccessful (for example, the number is unavailable or invalid) and no HLR MCCMNC was received
 - 3 - the HLR request was unsuccessful (for example, no response was received from the provider or the provider response code is unknown to the System)
- Note that System codes are not assigned to all HLR providers, and also assigning a value for different HLR providers is based on different fields from their responses. For more detailed logic description, contact the Alaris technical support team and provide the MNP-485 code and the HLR source type (for example, *netnumbercid*).

Clients

Extra fields to be added:

- *message_id*: the identifier of the request (assigned by the client)
- *rate*: the client rate from the corresponding HLR client product
- *wireless*: if the number was found in the MNP database (a boolean value - 0: no, 1: yes). Applicable to the *mnf* source only
- *cached*: if the result was obtained from the cache (a boolean value - 0: no, 1: yes)
- *present, roaming*: if the number is present/if the number appears in roaming, respectively, over the HTTP protocol. To add information for an ENUM response, contact the Alaris technical support team and provide the code HLR-23. If the flags cannot be recognized from the source HLR response, the returned value will be NA. Other possible values: *yes*, *no*. The logic for fetching the *present* and *roaming* fields is added for some types of HLR sources and described below:
 - *mitto* - *present* is fetched from the *absent* field (if *absent=true*, then *present=no*, if *absent=no*, then *present=yes*); *roaming* is fetched from the *roaming* field
 - *tyntec* - *present* is fetched from the *presence* field, *roaming* - from the *roaming* field
 - *xconnect* - *present* is taken from the *ns* field (if *ns=000* then *present=yes*; if *ns=001*, or *002* - *present=no*, if *ns=003* then *present=NA*), the *roaming* field is not recognized
 - *tmtlive* - *present* is taken from *present*, the *roaming* field is not recognized
 - *infobip* - *present* is recognized from {*status*}{*groupName*} (if *groupName=DELIVERED*, then *present=yes*, if *groupName=UNDELIVERABLE*, then *present=no*), *roaming* - from the *roaming* field
 - *netnumber_mnis* - *present* is taken from *status*, the *roaming* field is not recognized
 - *hgc2_enum* - *present* is taken from *pres*, the *roaming* field is not recognized
 - *hlrlookup_v2* - *present* is taken from *live_status* (if *live_status=LIVE*, then *present=yes*; if *live_status=DEAD*, or *ABSENT_SUBSCRIBER*, then *present=no*; otherwise *present=NA*), the *roaming* field is not recognized
 - *jtnav_enum_gr* - *present* is taken from *valid* (*present=yes* if *valid=true*; *present=no* if *valid=false*; the value NA will be assigned for any other values of the *valid* field), the *roaming* field is not recognized
- *message*: contains extended information in regard to the requested number (for example, "Not valid response" if the HLR source response was not parsed by the HLR module, or "Vendor not found" if no HLR dipping rule was found suitable), may be absent from the response in case of success
- *context_log*: full raw HLR source response proxied from the HLR source alongside internal information of parsing
- *providerResponseCode*: *hlrResponseCode* from the HLR source response
- *raw_response*: serves to proxy the HLR vendor's raw response to clients
- *aux_sources*: serves to transfer the full list of queried HLR vendors. If not specified, only the final queried vendor will be returned (for example, if the previous vendor did not return the MCCMNC or if the *Reroute and keep MCCMNC* flag was enabled).

Clients

- *error*: HLR provider's error code
- *trash, unallocated, is_rcs_capable, is_wa_capable* (boolean values): applicable if the HLR source is MNP and the *Custom MNP* parameter is set to a corresponding value
- *language_name, language_confidence, antifraud_match, message_category*: applicable for AITC integration; language name (for example, *en*), reliability of language definition (for example, *0.9* which stands for 90% of reliability), spam probability, and message category (for example, *OTP*) correspondingly

Limitations:

- *Max sessions*: the default value is 1000. The maximum number of session per client.

The following parameters limit the number of requests sent by HLR reselling clients:

- *Request limit* (disabled by default): the maximum number of incoming requests within the period defined by the *Period type* and *Period number* fields. Possible values: from 1 to 100,000,000. If the field is left empty, no limits are applied.
- *Period type* (available if *Request limit* is enabled): the type of time period; possible values: *hour, day, week*.
- *Period number* (available if *Request limit* is enabled): the number of periods based on the *Period type*; possible values: from 1 to 500.
- *Effective from* (available if *Request limit* is enabled): the date and time from which the request count will start.

For example, if the *Request limit* is set to 1000 and *Period number* is 3 with *Period type* = *day*, then data will be accumulated over a span of 3 days starting from the *Effective from* date. Suppose the *Effective from* date is set to 2020-01-01 00:00:00; in this case, the counter will reset at 2020-01-04 00:00:00 — and then every three days thereafter.

NOTE: Requests processed before the configuration was set are not taken into account when calculating the limits. **Important:** All client requests are taken into account, for which a response with an error code other than 4xx was returned. When the limit is exceeded, subsequent requests will be rejected with the error message: "Request limit exceeded."

History logs

5 History logs

The *History logs* interface serves to track changes applied by users in the MNP configurator interfaces. By default, change logs are available for the past seven days. To change the number of days contact the Alaris technical support team and provide the code MNP-235.

Click *Filter* to filter records in the logs as appropriate.

History logs

ana@alarislabs.com

Total: 94

Filter

Refresh page

ID	ACTION ID	TABLE NAME	ACTION TYPE	RECORD COUNT	ACTION TIME	USER
5767	019ce0d8-3187-7745-bfa7-260462287fc3	hlr_simulation_rules	INSERT	2	2026-03-12T06:59:50.791085Z	OB
5769	019ce0d9-2954-7f86-9b87-06c34b97f499	hlr_simulation_rules	INSERT	2	2026-03-12T07:00:54.228709Z	OB
5771	019ce0d9-a216-70f0-a851-4601b4cf3429	hlr_simulation_rules	INSERT	2	2026-03-12T07:01:25.142354Z	OB
5773	019ce0d9-e98c-739c-b9bc-1a71799207bf	hlr_simulation_rules	INSERT	2	2026-03-12T07:01:43.436323Z	OB
5775	019ce0dd-a1a6-7500-a0d0-c7b4efb9d9bb	hlr_simulation_rules	INSERT	2	2026-03-12T07:05:47.174973Z	OB
5777	019ce0de-7aff-7c59-80e7-0ecf43681f39	hlr_simulation_rules	INSERT	2	2026-03-12T07:06:42.815679Z	OB
5779	019ce0de-e01f-74d0-aa42-ca02b200fcfe	hlr_simulation_rules	INSERT	2	2026-03-12T07:07:08.704444Z	OB
	019ce0e0-b436-7302-				2026-03-	

Showing 1 - 50

Prev 1 2 Next

Show rows 50

History logs

The table contains the following columns:

- *ID*: sequential internal identifier of the log entry.
- *ACTION ID*: a unique string identifying the specific operation.
- *TABLE NAME*: the System table impacted by the change. The table name reflects the interface in which the change was made. For example, the *hlr_settings* table corresponds to the [Settings](#)^[36] interface.
- *ACTION TYPE*: the nature of the change. Possible values are:
 - *INSERT*: an entity has been added.
 - *UPDATE*: an entity has been modified.
 - *DELETE*: an entity has been deleted.
 - *Import file*: rate import in the [Import rates](#)^[40] interface.
 - *Import config*: the MNP proxy configuration file has been updated (*Import configuration from file* function of the [Settings](#)^[36] interface).
- *RECORD COUNT*: number of records affected during the action.

History logs

- *ACTION TIME*: date and time of the logged event.
- *User*: the login of the user that performed the changes.

Hover over a record and click the information icon  to view the details as illustrated below. The changes are highlighted in color.

Action ID 019ce0d8-3187-7745-bfa7-260462287fc3

Before

author_id: 4
created_at: 2026-03-12T06:59:50.791085+00:00
dipping_rule:
id: 101
prefix: 808

After

author_id: 4
created_at: 2026-03-12T06:59:50.791085+00:00
dipping_rule:
id: 101
prefix: 808
product_id: 32321
product_name:
sender_id:
status: inProgress

Action ID 019ce0d8-3187-7745-bfa7-260462287fc3

Before

author_id: 4
created_at: 2026-03-12T06:59:50.791085+00:00
dipping_rule:
id: 101
prefix: 808

After

author_id: 4
created_at: 2026-03-12T06:59:50.791085+00:00
dipping_rule: OB_2
id: 101
prefix: 808

Close

Log record details

Changes are highlighted in color. For the *DELETE* action, the *Before* section is filled, containing parameters of the entity that was deleted. For *INSERT*, only the *After* section is filled showing what has been added; for *UPDATE*, both *Before* and *After* are filled.

Reselling API

6 Reselling API

This section details an API for working with third-party systems for HLR reselling. The supported protocols are HTTP (GET method) and ENUM.

6.1 HTTP

1. HTTP example of a request (values highlighted in red must be replaced by actual values)

```
http://<HLRdomain>:port/hlr/mccmnc_request?  
login=<userlogin>&password=<userpass>&dnis=<dnis>&message_id=1&sender_id=Facebook&noCache=1
```

where

HLRdomain: HLR domain name

port: the HTTP listening port. The default value is 42081, can be overridden by internal configuration

userlogin: login

userpass: password

dnis: destination addresses in e.164 format

message_id: request ID, may come in handy for troubleshooting purpose

sender_id: sender ID, optional parameter

noCache: optional parameter, specify as 1 if a non-cached response must be obtained

2. Response format

The fields returned in the response depend on the client's configuration as the list is configured with the help of the *Response fields* field. The fields enabled by default are described below.

2.1. Successful response:

```
{  
  "mccmnc": "234001",  
  "result": 0,  
  "ported": 0,  
  "source_name": "some_mnp",  
  "source_type": "mnp",  
  "dnis": "34567891001",  
  "login": "login"  
}
```

where

"mccmnc": 6-digit HLR MCCMNC

Reselling API

"result": shows if the result is successful (0 – yes, other values - no)

"ported": shows if the number is ported (0 - no, 1 - yes)

"source_name": HLR source name

"source_type": HLR source type

2.2. Unsuccessful response

An unsuccessful response does not contain an HLR MCCMNC

```
{
  "result": -1,
  "ported": 0,
  "source_name": "some_mnp",
  "source_type": "mnp",
  "dnis": "34567891001",
  "login": "login"
}
```

where

result: the following values can be returned for an unsuccessful response:

- -1 - No HLR MCCMNC found for the requested destination address (a non-empty response was received)
- -2 - No HLR rule was considered suitable or DNIS length is exceeding
- -3 - No valid MCCMNC found for the requested destination address (for example, an empty response was received) or no translation rule is found for the returned error code/status
- -4; -6; -7 – System timeout
- -5 – the number is blacklisted
- -10 – the number's MCCMNC could not be found in the System reference book and the *block_requests_with_no_mccmnc* internal flag is enabled

6.2 ENUM

1. ENUM example of a request (values highlighted in red must be replaced by actual values)

The default request (if the checkbox *Use Login to authorize client* is disabled in the [Clients](#) ¹⁶ interface):

```
dig @<hlrAddress> -t NAPTR <reverseDnis>.<dummyDomainName> -p <port> +short
```

where

<hlrAddress>: the HLR server IP address

<reverseDnis>: the destination address number in reverse, separated by dots. For example, if the original number is 123456789, the **reverseDnis** will be 9.8.7.6.5.4.3.2.1

<dummyDomainName>: any domain name. For instance, e164enum.int.alarislabs.com

Reselling API

<port>: the ENUM listening port. The default value is 5053, can be overridden by internal configuration

If the checkbox *Use Login to authorize client* is enabled in the [Clients](#) ¹⁶ interface, the request will look as follows:

```
dig @<hlrAddress> -p <port> IN NAPTR <reverseDnis>.<login>.<dummyDomainName>
```

where **<login>** is the client login name set in the [Clients](#) ¹⁶ interface, and the other parameters are similar to the default request.

Example:

```
dig @127.0.0.1 -p 5053 IN NAPTR 2.3.3.4.0.9.8.5.9.5.8.3.admin.enum.int.alarislabs.com
```

The fields returned in the response depend on the system-wide internal configuration. Contact the Alaris technical support team to learn more or to configure a response template.

2.1. Successful response

;; ANSWER SECTION:

```
2.8.1.0.1.4.2.2.6.2.1.2.e164.arpa. 0 IN NAPTR 100 10 "U" "E2U+pstn:tel"
"!^(.*)$!country=null;operator=null;mcc=604;mnc=001;ported=false;err=0!" .
```

Possible values also include:

operator - network name

provider_response_code - hlrResponseCode

dnis - destination address

isCached - if the information was obtained from cache

rate - rate from the HLR client product

source_name, *source_type* - name and type of the HLR source

wireless, *present*, *roaming*, *trash*, *unallocated*, *is_rcs_capable*, *is_wa_capable* - boolean flags identical to the ones described in the *Response fields* parameter on the [Clients](#) ¹⁶ page

2.2. Unsuccessful response

A response does not contain an HLR MCCMNC:

```
3.2.4.3.6.0.4.6.2.0.2.e164.arpa. 0 IN NAPTR 100 10 "U" "E2U+pstn:tel"
"!^(.*)$!country=null;operator=null;mcc=null;mnc=null;ported=false;err=-3!" .
```

The *err* values are the same as for the *result* values of the [HTTP reselling API](#) ²⁴.

6.3 Camara

To send requests and retrieve information over HTTP, the client must follow the API described below (values highlighted in red must be replaced by actual values).

Step 1. Obtain a token for a specific service

Send the following request, replacing *127.0.0.1* with the IP address or domain name of the MNP server, *LOGIN* and *PASS* with the client credentials, and *SCOPEVALUE* with one of the supported scope values:

Reselling API

```
curl "http://127.0.0.1:42081/oauth2/v1/token" -d
"grant_type=client_credentials&scope=SCOPEVALUE" -H 'Content-Type: application/x-www-
form-urlencoded' -u "LOGIN:PASS"
```

Supported values of *SCOPEVALUE*: *sim-swap:retrive-date*, *sim-swap:check*, *number-verification:verify*, *number-verification:device-phone-number*, *kyc-match:match*.

The response contains a token, for example:

```
{"access_token":"TOKEN","token_type":"Bearer","expires_in":3600,"scope":"SCOPEVALUE"}
```

The issued token is valid for 3600 seconds by default. To change the token lifetime, contact the Technical support team and provide the code MNP-981.

Step 2. Query a specific service

The following examples show how to query each supported service. In all examples, replace *TOKEN* with the token value obtained in Step 1.

a) Sim swap (check)

```
curl -X POST 'http://127.0.0.1:42081/sim-swap/v2/check' -H 'authorization: Bearer TOKEN'
-H 'content-type: application/json' -H 'x-correlator: 8f5bd6bc-6f54-4d93-a895-
354f50fda6dd' -d '{"phoneNumber": "+346661113334", "maxAge": 24}'
```

Parameters: *x-correlator* — unique request identifier; *phoneNumber* — the number to check; *maxAge* — the period in hours during which the SIM card change is checked.

Response: { "swapped": true }

b) Sim swap (retrieve date)

```
curl -X POST 'http://127.0.0.1:42081/sim-swap/v2/retrieve-date' -H 'authorization: Bearer
TOKEN' -H 'content-type: application/json' -H 'x-correlator: 8f5bd6bc-6f54-4d93-a895-
354f50fda6dd' -d '{"phoneNumber": "+346661113334"}'
```

Response contains the date of the last SIM card change. If the SIM card has never been changed, the date will be *null*:

```
{ "latestSimChange": "2024-09-18T07:37:53.471829447Z" }
```

c) Number verification (verify)

```
curl -X POST 'http://127.0.0.1:42081/number-verification/v2/verify' -H 'authorization:
Bearer TOKEN' -H 'content-type: application/json' -H 'x-correlator: nv-device-phone-001'
-d '{"phoneNumber": "+346661113334"}'
```

Response: { "devicePhoneNumberVerified": true }

d) Number verification (phone number)

```
curl -i -sS -X GET 'http://127.0.0.1:42081/number-verification/v2/device-phone-number' -H
"Authorization: Bearer TOKEN" -H 'x-correlator: nv-device-phone-001'
```

The *x-correlator* value must match the one sent in the corresponding *number-verification/v2/verify* request.

Response: { "devicePhoneNumber": "+346661113334" }

e) Know your customer (data match)

```
curl -X POST 'http://127.0.0.1:42081/kyc-match/v0.3/match' -H 'authorization: Bearer
TOKEN' -H 'content-type: application/json' -H 'x-correlator: kyc-match-002' -d
'{"phoneNumber": "+346661113334", "idDocument": "66666666Q", "name": "Federica Sanchez"
}
```

Reselling API

```
Arjona", "givenName": "Federica", "familyName": "Sanchez Arjona", "birthdate": "1978-08-22", "email": "federica@example.com", "postalCode": "28001", "country": "ES" }'
```

The *phoneNumber* field is mandatory. At least one optional field must also be provided.

Response (may vary depending on the service response):

```
{ "idDocumentMatch": true, "nameMatch": true, "givenNameMatch": true, "familyNameMatch": true, "birthdateMatch": true, "emailMatch": false, "postalCodeMatch": true, "countryMatch": true }
```

Dipping rules

7 Dipping rules

The page serves to configure HLR request rules. That is, which HLR services are to be used for number verification. Configuration can be prefix-, sender ID- and client-based.

The main table shows available rules that can be filtered with the help of the *Name*, *Choice list*, *Prefix list* and *Product list* filters.

NOTE: In the *Prefix list* field, exact match search is performed. Masks (through %) are allowed for substring search. Below are examples of acceptable values:

- 34% - search for rules where the prefix starts with 34
- %34 - search for rules where the prefix ends with 34
- %34% - search for rules containing the prefix 34
- 34%,1%2,86 - search for rules where the prefix starts with 34, or starts with 1 and ends with 2, or matches "86"

To edit a rule, hover over the record and click the Edit  button. To remove a rule, use the Delete  button. To clone a rule, click .

Dipping rules

ana@alarislabs.com

Total: 1  Filter

 Refresh page  Export  Add new rule

ID	NAME	CHOICE LIST	PREFIX LIST 	PRODUCT LIST	PRODUCT TYPE
853	KATETEST		1 4 5 6 7 8 20 21 22 24 25 26	Vendor (25056)	

Dipping rules

To add a new rule, click the *Add new rule* button. Configure the following fields in the window that appears:

Dipping rules



 **Create new rule**

Name

Spain TMT

Prefix list *

34511120-34511125 

Sender ID list

info  exceder  perfection 

Products inclusive list 

Product list

15993 - 15993_Client_2 (28471)  15993 - 15993_Vendor (28470) 

Product type

hlr

Product names

1  2  adv2 

Choice #1

Source *
1

Share
80 %

Delete 

Source *
CNsource

Share
20 %



+ Add source

Enable LCR
No

+ Add choice

 Reset

 Submit

Close

30

Intelligent Control & Monetization for Telecom Business



Dipping rules

Create new rule

- **Name:** name of the rule, for example: *Spain TMT*. Mandatory parameter
- **Prefix list:** list of numeric prefixes for which the rule will be activated. The field is mandatory. It is possible to specify a range of prefixes, for example: *34511120-34511125* which includes prefixes *34511120, 34511121, 34511122, 34511123, 34511124, 34511125*. To add a value to the list, press *Enter*.

NOTE: The length of the range beginning and end must coincide. That is, it is not possible to set a range as *3451112-34511125*. Additionally, an exception can be specified to exclude a particular prefix. For example:

34

!34511120

- **Sender ID list:** list of sender IDs for which the rule will be activated. By default all sender IDs are included in the rule. Alpha, alphanumeric and numeric sender IDs are allowed. Hyphens, blank spaces and dots are allowed. Regular expressions are allowed. When added, regular expressions are highlighted in blue. The System detects a regular expression by the *^* character opening the expression and *\$* character closing it.

Possible options are:

- *^d*\$* - any number of digits (0 or more)
- *^d+\$* - one or more digits
- *^d{n}\$* - exact number of n digits
- *^d{m,n}\$* - from m to n digits

Combinations with alternatives are also supported, for example: *^(574\d{7}|565\d{8})\$* - the Sender ID is considered appropriate if it starts with 574 followed by exactly 7 digits, or with 565 followed by exactly 8 digits.

Please note that if both exact values and regular expressions are added within a single HLR rule, the exact values are searched first.

To add a value to the list, click *Enter*.

- **Products inclusive list:** toggle by clicking the  button. When enabled, the *Product list* field will appear.
- **Product list** (available if the *Products inclusive list* toggle is enabled): specify the list of client products for which the rule will be activated. All products are included by default. Select required products to enable the rule for particular client products. The product ID is displayed in brackets after the product name.
- **Product type:** select the client product type(s). Possible values include *hlr* and *sms*.
- **Product names:** select the appropriate client product name(s).
- **Choice list:** select the HLR provider names. Several providers can be set up in a specific order - once they are selected, the drag&drop window appears to configure the order. In this case, if the first provider does not supply an HLR MCCMNC (or it is impossible to parse it from the provider's response), the next-in-line provider will be inquired. The user can also select HLR sources from the *Choice list* with a specified probability. By default, each choice is defined with a probability of 100%. To add a probability distribution, click the *Add source* button, add

Dipping rules

additional sources and in the *Share* fields specify the probability for all added sources (the total probability must be 100%).

NOTE: Share-based HLR routing cannot be created for LCR rules or choices

NOTE: Overlapping rules cannot be created. For example, if 2 rules with empty sender IDs and product IDs (which is "All") for prefix 852 are created, the error will be shown: "Prefix collision error".

- *Enable LCR:* select to enable LCR routing for more flexible selection of HLR sources. When enabled, configuring the list of HLR sources (choices) is optional. To learn more, refer to [How to configure HLR LCR routing](#) ^[32] section.
-

NOTE: Search for suitable HLR rules has the following prioritization: product ID, product name, sender ID, prefixes.

NOTE: By default, only three HLR sources are used. To change the number of sources, contact the Alaris technical support team and communicate the code MNP-636.

Click the *Submit* button to save the changes or the *Close* button to disregard them. The *Reset* button will reset the configured changes.

7.1 How to configure LCR-based HLR routing

LCR routing stands for least-cost rates routing - that is, LCR functionality allows selecting HLR sources with the least cost to query a number. The configuration process consists of the following steps:

1. Configure currency exchange rates in [Exchange rates](#) ^[38], if HLR sources have different currencies or their currencies differ from the System one (marked as System currency in [Currencies](#) ^[37]).
2. Add manually or import rates from a file on the [Import rates](#) ^[40] page for HLR sources.
3. Create an HLR rule ([Dipping rules](#) ^[29]), select the *Enable LCR* flag and either leave the list of HLR sources empty, or indicate particular ones in the required order.

The order of HLR sources will be defined in accordance with the following logic:

1. A dipping query for a destination number is received.
2. By default, the destination address' MCCMNC is initially defined in accordance with Alaris SMS Platform's e.212/e.164 reference book. This match will be used for selecting rates for LCR only.

NOTE: The reference book is uploaded to the MNP proxy's in-memory DB once a day. If a change has been made to it recently, MNP proxy may still operate the previous info. To apply the change, contact the Alaris technical support team.

3. If MCCMNC could not be defined, rates for MCC Rest of the world will be searched (by default, it is MCC 777). Alternatively, it is possible to configure rejection for such dipping request. For this, contact the Alaris technical support team and request them to enable the internal parameter *block_requests_with_no_mccmnc*.
4. Once the MCC/MCCMNC is defined, rates for it are searched in the HLR sources. If the *Enable LCR flag* has been selected and the list of HLR sources has been left empty, all sources with rates will be used as choices.

NOTE: The System searches for the lowest price regardless of the accuracy of the MCC or MCCMNC match (for example, if there is source 1 with price 0.1 for MCC 304 and source 2 with price 0.2 for MCCMNC 304001, where 304001 is a more accurate match for the number, source 1 will still be selected).

Dipping rules

If an exact list of HLR sources is specified in the rule and the LCR mode is enabled, the System will use sources from the list for which rates on the required MCC/MCCMNC exist, sorting them from the least cost to the highest one.

NOTE: By default, only three HLR sources are used. To change the number of sources, contact the Alaris technical support team and communicate the code MNP-636.

Camara API Rules

8 Camara API Rules

Camara API allows retrieving additional subscriber information (over HTTP) such as:

- **sim swap** — retrieves the date of the subscriber's last SIM card change.
- **number verification** — verifies the availability and validity of a phone number.
- **know your customer match** — verifies subscriber data against reference information.

The *Camara API Rules* interface serves to configure request rules for Camara sources. Camara rules operate independently of standard HLR dipping rules and do not support fallback or rerouting of requests. Accordingly, when a request is received that must be processed through Camara services, standard dipping will not be performed.

ID	RULE TYPE	CLIENT	PREFIX	SOURCES	CREATED BY	LAST UPDATED
1	Know your customer (data match)	tess	1234,123	camara_source	tese	2026-06-26T06:17:47.318693Z
2	Sim swap (retrieve date)	tess12	1337	sim_swap_camara_test	tese	2026-06-26T06:20:59.100545Z
3	Sim swap (check)	cat	14	camara_sim_swap	tese	2026-06-26T06:22:09.899158Z
4	Number verification (verify)	catt	1000	camara_num_varify_test	tese	2026-06-26T06:24:26.845894Z

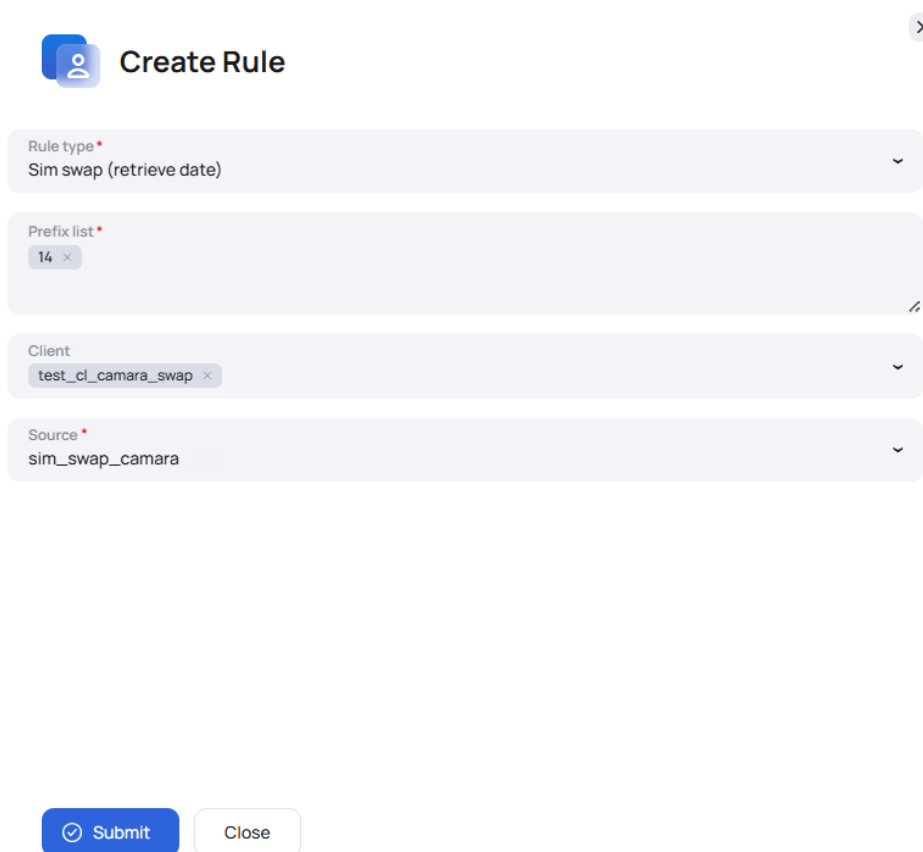
Camara API rules

The *Camara API Rules* interface contains the following elements and controls:

- *Filter*: filters the list of rules by specified criteria.
- *Refresh page*: reloads the page to display the most recent data.
- *Export*: exports the list of rules in a MS Excel file.
- *Create Rule*: opens the rule creation form.
- Table of rules that displays the following columns:
 - *ID*: the unique identifier of the rule.
 - *Rule type*: the type of the rule, indicating which Camara service will be queried.
 - *Client*: the name of the client associated with the rule.
 - *Prefix*: the list of phone number prefixes or the phone numbers used for rule selection.
 - *Sources*: the sources configured for processing requests.
 - *Created by*: the name of the user who created the rule.
 - *Last updated*: the date and time of the last modification.

Creating a Rule

Camara API Rules



Create Rule

To create a new rule, click *Create Rule*. In the form that opens, specify the following parameters:

- *Rule type*: select the type of rule, that is, which Camara service will be queried. The following values are available:
 - *Sim swap (check)*: verifies whether the subscriber's SIM card was changed within a specified period.
 - *Sim swap (retrieve date)*: retrieves the date of the last SIM card change.
 - *Number verification (verify)*: verifies the phone number provided in the request.
 - *Number verification (phone number)*: retrieves the phone number for which a Number verification (verify) request was received, based on the correlator ID sent in the same step.
 - *Know your customer (data match)*: verifies subscriber data against reference information.
- *Prefix list*: the list of prefixes and/or phone numbers used for rule selection. Type a value and click Enter to add it to the field.
- *Client*: the client name (the list contains the clients for which the corresponding Camara service has been selected on the [Clients](#) page in the *OAuth Scopes* field). If left empty, all clients are selected.
- *Source*: the source for the request.

Once all parameters are filled in, click *Submit* to confirm or *Close* to cancel.

Settings

9 Settings

The *Settings* interface contains three sub-interfaces: [Settings](#)^[36], [Currencies](#)^[37] and [Exchange rates](#)^[38].

9.1 Settings

The page serves to configure global settings (that are applied to all HLR services and clients).

- *Default connection timeout*: timeout (in seconds, default value is 2) to establish a connection with the HTTP HLR service. Positive integer values are allowed
- *Default request timeout*: timeout (in seconds, default value is 2) to expect a response from the HTTP HLR service (if a value is not configured in the *HTTP request timeout* parameter of [Sources](#)^[7]). Positive integer values are allowed
- *Concurrent request timeout*: timeout (in seconds, default value is 5) to wait for concurrent requests. That is, if several requests to the same number have been received within the timeout, only one request will be dipped through the HLR service. If it provides an HLR MCCMNC, the MCCMNC will be used as cached values for the rest of requests, otherwise the requests will be dipped through the HLR service as well. Positive integer values are allowed
- *Enterprise API hostname*: the address to fetch data (for example, HLR client product IDs) from the database to show it (actual names instead of IDs) in the MNP configurator interfaces such as [Dipping rules](#)^[29] or [Clients](#)^[16]. Applicable to the MNP Configurator System which is installed separate from the main Alaris System.

Example: <http://localhost:3006/eapi/>

- *DNIS limit*: the number of digits allowed for the length of destination addresses. For example, if specified as 13, destination addresses of 13 or less digits are allowed, otherwise, *result=-2* will be returned in a response. If set as 0, no limitation is applied.
- *Check global rules for products*: when deselected, Dipping rules with "All" client products are not checked if there is at least one client-based rule. Enabled by default.

Example: suppose there are 2 rules:

Rule 1 for prefix 34 and all client products

Rule 2 for prefix 852 and client product ID 45

If the checkbox is deselected and a request from client product ID 45 is received, only Rule 2 can be used (if the request is intended for another prefix - for example, 34, no rule will be used). If the checkbox is selected, both rules can be used (but Rule 2 has a priority over Rule 1).

- *Check global rules for Sender IDs*: when deselected, Dipping rules with "All" sender IDs are not checked if there is at least one client-based rule. Enabled by default.

Example: suppose there are 2 rules:

Rule 1 for prefix 34 and all sender IDs

Rule 2 for prefix 852 and sender ID "Facebook"

If the checkbox is deselected and a request from sender ID Facebook is received, only Rule 2 can be used (if the request is intended for another prefix - for example, 34, no rule will be used). If the checkbox is selected, both rules can be used (but Rule 2 has a priority over Rule 1)

Save the changes with the help of the *Submit* button.

Settings

- *Import configuration from file*: option usually used by the Alaris technical support to re-apply the internal file of MNP proxy.

9.2 Currencies

The *Currencies* interface serves to manage currencies. The functionality comes instrumental when using LCR in dipping rules.

NOTE: Created currencies are not synchronized with currencies configured in Alaris SMS Platform.

Currencies

iana@alarislabs.com

Total: 11
Filter
Refresh page
+ Add new currency

CURRENCY NAME	SYMBOL
 BRL Brazilian Real	R\$
 CNY Chinese Yuan	¥
 EUR Euro	€
 INR Indian Rupee	₹
PA Past	過去
 RUB Russian Ruble	₽
TO Today	=+_+=
 TRY Turkish Lira	₺
 AED United Arab Emirates Dirham	د.إ.
 USD US Dollar System currency	\$

Showing 1 – 11
Show rows 25

Currencies

The interface displays the list of available currencies. One of them is indicated as the System currency. By default, the System currency is set the same as the system currency of the main (transit) System with the help of Alaris technical support team.

To add a new currency, click *+Add new currency* and configure the fields as illustrated in the figure below. Click *Submit* to save the changes.

Settings

 Add new currency

Currency name
US Dollar

Char code
USD

Symbol
\$

Icon
icon-US

Reset

Submit

Close

Add new currency

9.3 Exchange rates

The *Exchange rates* interface serves to manage the exchange rates of currencies available in the System. The functionality comes instrumental when using LCR in dipping rules.

NOTE: Exchange rates are not synchronized with exchange rates configured in Alaris SMS Platform.

Exchange rates

iana@alarislabs.com

Total: 221  Filter  Refresh page  Add new exchange rate

CURRENCY	DATE	EXCHANGE RATE
 EUR Euro	10.10.2025	1
 TRY Turkish Lira	05.10.2025	12.01
 EUR Euro	03.10.2025	0.85
 RUB Russian Ruble	09.10.2025	9.32
 RUB Russian Ruble	10.10.2025	9.32
 TRY Turkish Lira	06.10.2025	12.01
 TRY Turkish Lira	07.10.2025	12.03
 TRY Turkish Lira	08.10.2025	12.5
 TRY Turkish Lira	09.10.2025	12.01
 TRY Turkish Lira	10.10.2025	12.01

Showing 1 – 25

Prev 1 2 3 4 5 ... 9 Next

Show rows 25

Settings

Exchange rates

The interface displays the list of available exchange rates to the System currency.

To add a new exchange rate, click *+Add new exchange rate* and configure the fields as illustrated in the figure below. Click *Submit* to save the changes.

 **Add new exchange rate** ×

Currency*
Euro

Date*
07.11.2025

Exchange rate
1

 Reset

 Submit

Close

Add new exchange rate

Import rates

10 Import rates

The *Import rates* interface is intended for uploading rates to use LCR functionality of [Dipping rules](#)^[29]. It is managed by the *import_rates.read*, *import_rates.edit* [permissions](#)^[54]. The interface displays a table of imported rates, and allows uploading and deleting rates.

NOTE: Imported rates are not synchronized with rates uploaded in Alaris SMS Platform.

Import rates

na@alarislabs.com

Total: 30

Filter

Refresh page

Import rates

Delete rates

ID	SOURCE	MCCMNC	RATE
1040	CNsource	412	0.079
1037	CNsource	289	0.34
1038	CNsource	289067	0.34
1039	CNsource	412025	0.079
1041	CNsource	412040	0.48
1035	CNsource	289088	0.34
1036	CNsource	289068	0.34
1050	ddd	250003	1.2
1049	ddd	250002	1.2
1048	ddd	250001	1.2

Showing 1 - 25

Prev 1 2 Next

Show rows 25

Import rates

Click *Import rates* to upload rates. In the window that appears select the source and drop or upload the rate file, and click *Confirm*.

Import rates

×

Source

BRtmtnlive



Drop file here or [click to upload](#)

File formats .xls, .xlsx, .csv , max size 10 Mb

Close

Confirm

Import rates

Supported file formats are xls, xlsx, and csv (with the separator - ;). The file must contain the following columns: *rate*, *mcc*, where *rate* supports numbers (including fractional ones), and *mcc* contains 3 digits.

Import rates

Also an optional column *mnc* can be used (possible values are up to 3 digits). Other columns will not be uploaded. When uploading the file, it is required to specify the HLR source in the web interface to which rates will be applied. MCC 777 (rest of the world) is supported for uploading. File example:

rate;mcc;mnc

0.5;301;002

0.2;322;01

The uploaded rates will appear in the table of rates. The user can delete all rates of a particular HLR source using the *Delete rates* button. Number (prefix) and MCCMNC matching is performed using the internal reference book. The MCC/MCCMNC for rate selection are defined by the longest match of the number from the e.212/e.164 reference book of Alaris SMS Platform.

Direct dipping

11 Direct dipping

The *Direct dipping* interface serves to check the operability of HLR sources through the MNP proxy directly (that is, without verification of created dipping rules - to review configured HLR routing, refer to [Simulation](#) ⁴⁵).

Direct dipping				iana@alarislabs.com	
Total: 2		Filter	Refresh page	Export	+ New
ID	STATUS	NUMBERS 1236 X			USER
MD-79 CNsource	Completed	12360900555 12360300550	12360100555 12360100551	990111 12360100552	Tati
MD-80 CNsource	Completed	14350900555 12360300550	12360100555 12360100551	990111 12360100552	Tati

Direct dipping

To create a new dipping request, click *New*.

Create new request

Source*

CNsource X

☒ Ignore cache

Phone number

Enter values

Upload file

Type in or copy & paste data here*

13450900555
13450100555
1990
13450300555
13450300557
13450300558
13450300559
13450300550
13450100551
13450100552

Close

Create

Create new request

Configure the fields as shown in the figure above:

- *Source*: select the HLR provider(s).

Direct dipping

- *Ignore cache*: when selected, requests are sent directly to the provider. If deselected, the data is taken from the System cache.
- *Phone number*: choose one of the following ways to supply phone numbers:
 - Type in or copy and paste the numbers in the edit box (separated by line break)

or

- Select *Upload file* and drag&drop or select an Excel file with the numbers. The file must contain only one column containing phone numbers. Other columns, if any, will be ignored.

Click *Create* to start the dipping task. The task will appear in the *Direct dipping* table. The *Status* column of the table includes the following values:

- *Completed*: marked with a green circle, the task has been completed and at least one response from the HLR provider is considered successful (in terms of parsing the HLR provider's response - for example, if the provider returned a response, but it was not possible to recognize it, it is considered unsuccessful; also if a response does not contain an HLR MCCMNC, but recognizable, the task is considered successful)
- *In progress*: the task is being processed
- *Failed*: marked with a red circle, it was not possible to complete a task (due to an internal error), or the HLR provider did not manage to respond within the timeout (the result error code is -6, -7)

Hover over the task and click ⓘ to view its details.

Task MD-79

10	9	9	0	0	1
Total	Successful	Ported	Not ported	Cached	Errors

Go to results >

⚙️ Status	● Completed
📁 Source	CNsource
📅 Created	11.11.2025 11:24 AM
👤 User	Tatiana
🔗 Cache	Ignored
📄 File	numbers.xlsx

Restart
Export results

Task details

where *Total* stands for the total amount of verified numbers and *Successful* reflects the amount of numbers for which responses that were parsed were received.

Click next to the file name to download the Excel file with the phone numbers. Click to copy the task ID to the clipboard.

Click *Export results* to download an Excel table with the task results.

Click *Go to results* to view the task results.

Direct dipping

Task MD-79

Successful Errors

MCCMNC 9 0 Empty
Ported Not ported

Total: 9 [Filter](#) [Refresh page](#)

PHONE NUMBER	MCCMNC	PORTED	SOURCE	RESPONSE TIME
1 (236) 010-05-51	310001	✓	CNsource	0.12
1 (236) 010-05-52	310001	✓	CNsource	0.38
1 (236) 010-05-55	310001	✓	CNsource	0.16
1 (236) 030-05-50	310001	✓	CNsource	0.17
1 (236) 030-05-55	310001	✓	CNsource	0.17
1 (236) 030-05-57	310001	✓	CNsource	0.15
1 (236) 030-05-58	310001	✓	CNsource	0.18

Showing 1 – 9 [Show rows](#) 25

Task, Successful tab

The window contains two tabs: *Successful* and *Errors*.

The *Successful* tab shows the list of numbers, their MCCMNCs, status (ported or not ported), source and response time (in seconds)

The *Errors* tab shows the phone numbers, errors (for example, *MCCMNC not found*, *Number is blacklisted*, *Response time exceeded*) and log. Click *view* to display the log.

Task MD-80

Successful Errors

Total: 1 [Filter](#) [Refresh page](#)

PHONE NUMBER	ERROR	LOG
▲ 990111	Not valid response : {"MCC": "", "MNC": "", "ported": "0"}	View

Showing 1 – 1 [Show rows](#) 25

Task, Errors tab

Simulation

12 Simulation

The *Simulation* interface is designed to run simulation according to existing dipping rules, without actually querying HLR sources. Using this interface, the System owner can:

- Check whether dipping rules are configured correctly, whether there are any conflicting rules, and what HLR source will be selected for the dipping request.
- Test rules with various client request parameters.

Access to the interface is controlled by the permissions *simulations.read* and *simulations.edit*.

The *Simulation* interface contains a table with the history of performed simulations.

Simulation

ana@alarislabs.com

Total: 147

Filter

Refresh page

Export

Create

ID	STATUS	PREFIX	SENDER ID	PRODUCT NAME	DIPPING RULE
SI-169 anbe Retail Wholesale 3 2	Completed	3	2	anbe Retail Wholesale	Undefined
SI-167 anbe Retail Wholesale 3 2	Completed	3	2	anbe Retail Wholesale	Undefined
SI-166 777	Completed	777			KATETEST2
SI-165 Test	Completed	Test			Undefined
SI-164 KA2 302001	Completed	302001		KA2	Undefined
SI-163 Olga_new 3009876	Completed	3009876		Olga_new	PreficsTest1
SI-160 1345 79200600555 Sender666	Completed	79200600555	Sender666	1345	TestSender
SI-159 808	Completed	808			OB_2_noprod

Showing 1 - 50

Prev 1 2 3 Next

Show rows 50

Simulation table

The table includes the following columns:

- *ID*: a unique task identifier and description of the simulation task.
- *Status*: the current state of the task (*Completed*, *In progress*, *Failed*).
- *Prefix*: the destination address used for the simulation.
- *Sender ID*: the sender ID specified in the test.
- *Product name*: the name of the client product selected for the check.
- *Dipping rule*: the Dipping rule applied to the request.

The following actions are available when hovering over a record:

- **View** : opens the task details window. Click *Restart* to create a task with identical or similar parameters.

Simulation



Task SI-164

Undefined
Dipping rule

⚙️ Status	● Completed
📅 Created	17.03.2026 10:02 AM
👤 User	OB
🔑 Sender ID	
📄 Prefix	302001
📦 Product name	KA2
📦 Client product	—

Restart

Task details

- *Delete* 🗑️: removes the selected simulation from the table.

The following control buttons are located above the table:

- *Filter*: allows filtering the records by table column values and creation dates.
- *Refresh page*: updates the table data to display the latest task statuses.
- *Export*: allows exporting the current table data into .xlsx format.
- *Create*: opens the *New simulation* window to initiate a new test, as illustrated below.

Simulation

New simulation



Sender ID
PocoDi

Prefix
1

Product type
hlr



Product name
2



Client product
12345 - 12345 (27937)



Close

Create

New simulation



Response code translations

13 Response code translations

The *Response code translations* interface serves to configure HLR error codes translation rules. That is, it helps the System owner standardize various HLR providers' response codes (*hlrResponseCode/providerResponseCode*) and send unified values to HLR clients (given that the *providerResponseCode* field is selected in the [Clients](#)^[16] interface). The functionality comes in handy for reselling purposes.

The interface contains two sub-interfaces: [System codes](#)^[48] and [Translation rules](#)^[49].

13.1 System codes

The *System code* page provides the list of the System response codes that can be used in [Translation rules](#)^[49] and allows the System owner to add custom codes. By default, the following codes are available:

0 - Dipping successful, MCCMNC returned, provider does not indicate failure

1 - Dipping successful, MCCMNC returned, but provider indicates number unavailable

2 - Dipping failed, MCCMNC not returned, number invalid or fixed (not mobile network)

3 - Dipping failed, no response from provider (timeout) or unknown provider response code

The same values refer to the *system_response_code* field that can be sent to HLR clients.

System codes

kate@alarislabs.co..

Total: 4

Filter

Refresh page

Create system code

ID	CODE VALUE	TYPE	CODE DESCRIPTION	LAST UPDATED
1	0	default	Dipping successful, MCCMNC returned, provider does not indicate failure	01-12-2025 05:51:44
2	1	default	Dipping successful, MCCMNC returned, but provider indicates number unavailable	01-12-2025 05:51:44
3	2	default	Dipping failed, MCCMNC not returned, number invalid or fixed (not mobile network)	01-12-2025 05:51:44
4	3	default	Dipping failed, no response from provider (timeout) or unknown provider response code	01-12-2025 05:51:44

System codes

To add a new user (custom) code, click the *Create system code* button. Configure the following fields in the window that appears:

Response code translations



 **Create system code**

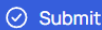
Code value

27

Code description

Absent subscriber



 Submit

 Close

Create system code

- *Code value*: the system code for a translation rule that can be configured in [Translation rules](#)^[49].
- *Code description*: a non-obligatory field for a code description, which will be available along with the *Code value* while configuring a rule in [Translation rules](#)^[49].

Click the *Submit* button to save the changes or the *Close* button to disregard them. The *Reset* button will reset the configured changes.

To edit a code, hover over the record and click the *Edit*  button. To remove a user code, use the *Delete*  button. Note that the System (default) codes cannot be edited or removed.

13.2 Translation rules

The *Translation rules* page serves to create a new translation rule or manage existing ones in order to unify codes sent to HLR clients that work over HTTP. The main table shows available rules that can be filtered with the help of the *Name*, *Status*, *Source type*, *Response codes* and *Source code* filters.

Response code translations

Translation rules						
Total: 2 Filter Refresh page + Create translation rule						
ID	NAME	STATUS	SOURCE TYPE	RESPONSE CODES	SYSTEM CODE	LAST UPDATED
2	Absent subscriber	Active	mitto	1,27,22	27	01-12-2025 15:55
3	Success	Inactive	tmtlive	Success,0,000	0	01-12-2025 16:11

Translation rules

To add a new rule, click the *Create translation rule* button. Configure the following fields in the window that appears:



Create translation rule

Name

Absent subscriber

☒ Active

Source type *

mitto

Response codes *

1 ×

27 ×

22 ×

System code *

27

Reset

☒ Submit

Close

Create translation rule

- *Name*: name of the translation rule.

Response code translations

- *Active*: select the flag to create an active rule; the flag is deselected by default.
- *Source type*: the HLR provider type, to which the rule will be applied. Note that if several HLR sources of the same type are created, the rule will be applied to all of them.
- *Response codes*: indicate the list of codes returned by the HLR provider. To add a value to the list, press *Enter*. Indicate the *null* value if absence of HLR provider code.
- *System code*: specify the System code that will be translated from *Response codes* to this value and sent to clients in the *system_response_code* field (given that the field is selected in the [Clients](#)  page).

Click the *Submit* button to save the changes or the *Close* button to disregard them. The *Reset* button will reset the configured changes.

To edit a rule, hover over the record and click the *Edit*  button. To remove a rule, use the *Delete*  button.

NOTE: Translation rules with custom (user) codes as *System code* have a priority over rules with System (default) codes in them.

Users

14 Users

The *Users* interface serves to create and edit user accounts.

Users

iana@alarislabs.com

Total: 52

Filter

Refresh page

Export

Add new user

LOGIN	EMAIL	NAME
Neverly	never@north.net	
Mary	mary@pocodinero.es	Mary Goode
John	john_doe@pocodinero.es	
Pat	pat@pocodinero.es	Patrick Best

Users

Click *Add new user* to create a new user record.



Add new user

×

Login *

Dreamer

Email *

dreamer@money.net

Password *

.....

👁

First name

Dreamer

Last name

Diminuendo

Page size

50

▼

Role permissions

User_vieweditpresent_role × user × tech_support_role_role × default_d

↻ Reset

✓ Submit

Close

Add new user

Configure the following fields:

- *Login*
- *Email*
- *Password*
- *First name*

Users

- *Last name*
- *Page size*: the number of records displayed per page
- *Role permissions*: select the role (created in [Roles](#)⁵⁴), the number of available permissions will be displayed in the field.

Click *Reset* to clear the fields. Click *Submit* to save the record and close the form. Click *Close* to close the form without saving.

Roles

15 Roles

The *Roles* interface serves to create and edit user roles and permission sets. By default, the following roles exist in the System: *tech* (full access) and *default* (same permissions except for access to the [Users](#)^[52] and [Roles](#)^[54] interfaces).

ROLE NAME	ROLE PERMISSIONS	USER
Norates.edit.past	rates.edit rates.read currencies.edit currencies.read settings.edit settings.read sources.edit sources.read providers.read rules.read rules.edit products.read clients.edit clients.read profile.edit roles.edit roles.read users.edit users.read import_rates.read import_rates.edit mdtask.read mdtask.edit mdresult.read	OB

Roles

Click *Add new role* to create a new role record.

Roles


Add new role
×

Role name *
user

Role permissions
les.read × roles.edit × settings.edit × sources.edit × sources.read × ^

☒ users.read
☒ users.edit
☒ roles.read
☒ roles.edit
☐ profile.edit
☐ clients.read

✓ Submit
Close

Add new user

Configure the following fields:

- *Role name*
- *Role permissions*: select appropriate permissions for the role from the drop-down list.

Click *Reset* to clear the fields. Click *Submit* to save the record and close the form. Click *Close* to close the form without saving.

15.1 Permissions

The following permissions are available in the System:

Permission	Description
users.read	View the Users ⁵² interface
users.edit	Edit records in the Users ⁵² interface
roles.read	View the Roles ⁵⁴ interface
roles.edit	Edit records in the Roles ⁵⁴ interface
profile.edit	Edit the user's profile in the Account settings ⁵ interface

Roles

Permission	Description
clients.read	View the Clients interface
clients.edit	Edit records in the Clients interface
products.read	View client products in the Clients and Dipping rules interfaces
providers.read	Get the list of HLR sources when creating or editing a source (in the <i>Type</i> field of the <i>Add new source</i> form)
rules.read	View the Dipping rules interface
sources.read	View the Sources interface
sources.edit	Edit records in the Sources interface
settings.read	View the Settings interface
settings.edit	Edit records in the Settings interface
currencies.read	View the Currencies interface
currencies.edit	Edit records in the Currencies interface
rates.read	View the Exchange rates interface
rates.edit	Add, delete and edit records in the Exchange rates interface with the current or a future start date
import_rates.read	View the Import rates interface
import_rates.edit	Import and delete rates in the Import rates interface
mdtask.read	View the Direct dipping interface
mdtask.edit	Create and delete tasks in the Direct dipping interface
mdresult.read	View dipping results (the <i>Go to results</i> button and a full error log) in the Direct dipping interface

Roles

Permission	Description
system_codes.read	View the Response code translations\System codes  interface
system_codes.edit	Create, edit and delete codes in Response code translations\System codes 
translation_rules.read	View the Response code translations\Translation rules 
translation_rules.edit	Create, edit and delete rules in Response code translations\Translation rules 
history_logs.read	View the History logs  interface
simulations.read	View the Simulation  interface
simulations.edit	Create, edit and delete simulations in the Simulation 
camara_rules.read	View the Camara API Rules  interface
camara_rules.edit	Create, edit and delete rules in the Camara API Rules 