PortaSwitch

External System Interfaces Guide

70
MAINTENANCE
RELEASE

WWW.PORTAONE.COM

Copyright notice & disclaimers

PortaSwitch® External System Interfaces, June 2020
Maintenance Release 70
V1.70.03

Please address your comments and suggestions to: Sales Department, PortaOne, Inc. Suite #408, 2963 Glen Drive, Coquitlam BC V3B 2P7 Canada.

Changes may be made periodically to the information in this publication. The changes will be incorporated in new editions of the guide. The software described in this document is furnished under a license agreement, and may be used or copied only in accordance with the terms thereof. It is against the law to copy the software on any other medium, except as specifically provided in the license agreement. The licensee may make one copy of the software for backup purposes. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopied, recorded or otherwise, without the prior written permission of PortaOne Inc.

The software license and limited warranty for the accompanying products are set forth in the information packet supplied with the product, and are incorporated herein by this reference. If you cannot locate the software license, contact your PortaOne representative for a copy.

All product names mentioned in this manual are for identification purposes only, and are either trademarks or registered trademarks of their respective owners.

Table of Contents

Preface

This document provides information for developers who want to interface their applications, platforms or systems with PortaSwitch®. The XML / JSON API interfaces are described in the **PortaBilling XML / JSON API Reference** and **PortaSIP XML / JSON API Reference**.

Where to get the latest version of this guide

The hard copy of this guide is updated upon major releases only, and does not always contain the latest material on enhancements that occur in-between minor releases. The online copy of this guide is always up to date, and integrates the latest changes to the product. You can access the latest copy of this guide at: **www.portaone.com/support/documentation/**.

Conventions

This publication uses the following conventions:

- Commands and keywords are given in **boldface**.
- Terminal sessions, console screens, or system file names are displayed in `fixed width font`.

⚠ **Exclamation mark** draws your attention to important actions that must be taken for proper configuration.

NOTE: Notes contain additional information to supplement or accentuate important points in the text.

⌛ **Timesaver** means that you can save time by taking the action described here.

👓 **Tips** provide information that might help you solve a problem.

⚙ **Gear** points out that this feature must be enabled on the Configuration server.

Trademarks and copyrights

PortaBilling®, PortaSIP® and PortaSwitch® are registered trademarks of PortaOne, Inc.

1. PortaBilling RADIUS interface

RADIUS protocol

PortaBilling® uses the RADIUS protocol as per **Remote Authentication Dial-In User Service (RADIUS), RFC 2865** and **RADIUS Accounting, RFC 2866**. By default, the PortaBilling® RADIUS server listens on port 1812 (UDP) for authentication requests and on port 1813 (UDP) for accounting requests.

RADIUS attributes

PortaBilling® uses a set of Cisco RADIUS VSAs (Vendor-Specific Attributes). For more information, please refer to the RADIUS Vendor-Specific Attributes Voice Implementation Guide at **Cisco website (http://www.cisco.com)**.

RADIUS attributes

To ensure compatibility with future releases of PortaBilling®, it is highly recommended that you follow the Cisco guidelines regarding which attributes must be contained in each authentication or accounting request. For more information, please refer to the RADIUS Vendor-Specific Attributes Voice Implementation Guide at **Cisco website (http://www.cisco.com)**.

Authentication / Authorization requests

Currently, the following attributes are required for correct processing of authentication / authorization requests:

Attribute	Description
NAS-IP-Address (required)	Specifies the IP address of the network access server that is requesting authentication
User-Name (required)	Indicates the name of the user being authenticated by the RADIUS server
Password	Indicates the user's password
h323-conf-id	The unique call identifier generated by the gateway. Used to identify the separate billable events (calls) within a single calling session
Called-Station-Id	The telephone number the user called. **Only for authorization requests**
h323-ivr-out	User-definable AV pairs sent from the NAS to the RADIUS server
Message-Authenticator	Used to authenticate and protect the integrity of Access-Requests in order to prevent spoofing.

Authentication / Authorization responses

Currently, the following standard attributes are used (A1 = authentication, A2 = authorization):

Attribute	A1	A2	Description
h323-billing-model	Y		Type of billing service for a specific session
h323-return-code	Y	Y	Return codes are the server's instructions to the network access server. The list of possible values are described in the *RADIUS Return Codes* section
h323-currency	Y	Y	Currency used with h323-credit-amount
h323-preferred-lang	Y	Y	Language to use for the audio prompt
h323-credit-amount	Y		Amount of credit (in currency) in the account
h323-redirect-number	Y		Optional Phone number to which the call is redirected
Message-Authenticator	Y	Y	Used to authenticate and protect the integrity of Access-Requests in order to prevent spoofing.

Accounting requests

Currently, the following attributes are required for correct processing of accounting requests:

Attribute	Description
NAS-IP-Address	Specifies the IP address of the network access server that is sending accounting requests
NAS-Port-Name	Indicates the name of the physical NAS port that is authenticating the user
User-Name	Indicates the name of the user for the accounting request
Calling-Station-Id	The telephone number the call came from
Called-Station-Id	The telephone number the user called
h323-setup-time	Indicates the setup time in NTP format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, year

h323-connect-time	Indicates the connect time in Network Time Protocol (NTP) format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, and year
h323-disconnect-time	Indicates the disconnect time in NTP format: hour, minutes, seconds, microseconds, time_zone, day, month, day_of_month, year
h323-conf-id	The unique call identifier generated by the gateway. Used to identify the separate billable events (calls) within a single calling session
h323-incoming-conf-id	The unique number for identifying a calling session on a NAS, where a session is closed when the calling party hangs up. The h323-incoming-conf-id number is used to: • Match the outbound and inbound call legs for a session on a particular NAS • Collect and match all records for multiple calls placed (within the bounds of a session) on the gateway
h323-disconnect-cause	Specifies the reason a call was disconnected
h323-call-origin	The NAS'es behavior in relation to the connection that is active for this leg. For example, answer on leg 1; originate on leg 2
h323-call-type	Protocol type or family used on this leg of the call
Acct-Session-Time	Indicates how long (in seconds) the user has received service
Acct-Status-Type	Indicates whether this Accounting-Request marks the beginning of the user service (Start) or the end Stop)
h323-ivr-out	User-definable AV pairs sent from the NAS to the RADIUS server
h323-remote-address	IP address of the remote voice equipment

RADIUS attributes for Internet billing

An Internet session consists of the following RADIUS requests:

- Authorize::NETACCESS
- Accounting::NETACCESS::Start
- Accounting::NETACCESS::Alive
- Accounting::NETACCESS::Stop

Authentication / Authorization requests

Currently, the following attributes are required for the correct processing of authentication / authorization requests:

Attribute	Description
User-Name (required)	This is the name of the user being authenticated by the RADIUS server.
Password	Encrypted password with PAP or CHAP authentication. Present only in an authentication request.
NAS-IP-Address (required)	Specifies the IP address of the network access server that is requesting authentication.
NAS-Port-Type	Indicates the type of the port on the network access server that is requesting authentication.
NAS-Port	Indicates the port on the network access server that is requesting authentication.
Service-Type	The type of service the user is requesting.
Framed-IP-Address	The IP address to assign to a user. It can be defined for a user under an account configuration or allocated by NAS from a pool of IP addresses.

Authentication / Authorization responses

The billing server can generate one of three responses to NAS:

- **Access Reject** – The user is unconditionally denied access to all requested network resources. Reasons may include failure to provide proof of identification or unknown or inactive user account.
- **Access Challenge** –Additional information such as a secondary password, PIN, etc. is requested from the user. Access Challenge is also used in complex authentication dialogues.
- **Access Accept** – The user is granted access. Once the user is authenticated, the billing server will check that the user is authorized to use the network service requested.

Accounting requests

The focus of accounting is to track both network resources usage and traffic characteristics.

The following attributes are used for the correct processing of accounting requests:

Attribute	Description
Acct-Status-Type	The type of accounting message: Start, Alive, Stop.
Calling-Station-Id	The user's telephone number for a Dialup session, MAC address for a PPPoE session and the IP address for a PPP session.
Called-Station-Id	The telephone number the user calls for a Dialup session, MAC address for a PPPoE session and the user IP address for a PPP session.
Acct-Session-Time	Defines how long the user is connected (for Alive requests) or was connected (for Stop requests).
Acct-Session-Id	A unique identifier for the session.
Acct-Input-Octets	The user's outgoing traffic (in bytes).
Acct-Output-Octets	The user's incoming traffic (in bytes).
Acct-Input-Packets	Indicates the number of received packets.
Acct-Output-Packets	Indicates the number of sent packets.
User-Name	Indicates the name of the user for the accounting request.
NAS-IP-Address	Specifies the IP address of the network access server that is sending accounting requests

PortaBilling® custom attributes

All custom attributes conform to the VSA syntax. Attributes received by PortaBilling® are entered into the `h323-ivr-out` attribute (VSA No. 1), encoded as `name:value` pairs. Attributes sent by PortaBilling® are entered into the `h323-ivr-in` attribute (VSA No. 1), encoded as `name:value` pairs. Unless otherwise specified, any custom attribute may only be included once for each request.

Attributes sent by a RADIUS client (gateway) to the PortaBilling® server

PortaBilling_Session

This allows you to control an account's login session (sessions are used to prevent simultaneous logins by debit accounts). This attribute may be used in the accounting record for a specific call leg in order to force the session to be unlocked. (For instance, to force a session unlock on the answer/VoIP call leg, instead of the customary answer/Telephony call leg). An optional `expires` parameter may control how long the session lock state is supposed to be held on BE (for example, to preserve the funds deposited for the session until the next radius message.)

Format:

```
PortaBilling_Session:<string>
PortaBilling_Session:lock=<lockvalue>[;expires=<seconds>]
```

Possible lock values:

Value	Description
nolock	Do not lock the account after the current request
unlock	Do not lock the account for the current session and unlock any other existing session for this account
relock	Lock the account for the current session and unlock any other existing session for this account
ignore	Do not set or remove any locks, and skip any lock checks

Example:

```
h323-ivr-out      = 'PortaBilling_Session:nolock'
```

PortaBilling_SessionEnv

This allows specifying the value of i_env for request from internal NAS node. For normal nodes with already defined i_env value, this attribute will be ignored.

Format:

```
PortaBilling_SessionEnv:<int>
```

Example:

```
h323-ivr-out = 'PortaBilling_SessionEnv:3'
```

PortaBilling_Ignore_Password

This allows the password check during authentication or authorization to be disabled.

Format:

```
PortaBilling_Ignore_Password:<YES|NO>
```

Possible values:

`YES` or `NO`.

Example:

```
h323-ivr-out      = PortaBilling_Ignore_Password:YES'
```

PortaBilling_CallbackHistory

This provides information about the other (second) call leg of a call, so that the first call leg can be billed properly (e.g. “do not bill the first leg if the second leg was not connected”).

This attribute may be present multiple times in a request. Only the last occurrence is actually used, while the others are ignored.

Format:

```
PortaBilling_CallbackHistory:<string>
```

Possible values:

Value	Description
START	The first call leg was started
ATTEMPT	The callback engine attempted to establish an outgoing call for the second leg
OK	The second call leg was successfully connected

If there is no OK entry, the call leg will be billed using a special rate plan associated in the product configuration with the "CALLBACK_FAIL" access code.

Example:

```
h323-ivr-out      = 'PortaBilling_CallbackHistory:START'
h323-ivr-out      = 'PortaBilling_CallbackHistory:ATTEMPT'
h323-ivr-out      = 'PortaBilling_CallbackHistory:OK'
```

PortaBilling_AccessCode

An access code for selecting a specific rate plan within a product. Although originally used to pass the "IVR access number" to billing, it may contain any string (e.g. "FIRSTCALL"). For Quintum compatibility, this attribute also has the synonym ACCESSCODE.

Format:

```
PortaBilling_AccessCode:<string>
```

Possible values:
Any string. The value passed will be compared to the values entered in the product's accessibility (case-sensitively).

Example:

```
h323-ivr-out      = 'PortaBilling_AccessCode:18001234567'
```

PortaBilling_Original_DNIS

This replaces the value of the Called-Station-Id attribute (i.e. the destination phone number used for call rating).

Format:

```
PortaBilling_Original_DNIS:<value>
```

where the value is a semicolon-separated list of name:value pairs.

Possible values:

E.164 phone number (a sequence of digits, * and/or # signs).

Example:

```
h323-ivr-out      = 'PortaBilling_Original_DNIS:19001234567'
```

PortaBilling_RestoreCacheLevel

This controls how the billing engine uses the account information retrieved during the authorization process for a request, in order to facilitate the processing of other requests. Its most common application is to disable this information for re-use, if for some reason information from the current request should not be applied to other call legs (e.g. in callback scripts).

Format:

```
PortaBilling_RestoreCacheLevel:<integer>
```

Possible values:

Value	Description
-1	Disable caching and reuse of account information.
0	Use the default policy.

Example:

```
h323-ivr-out      = ' PortaBilling_RestoreCacheLevel:-1'
```

PortaOne-Service-Type (formerly Service-Identifier)

This contains the identifier for a service in PortaBilling®.

Format:

```
PortaOne-Service-Type=<value>
```

Possible values:

A non-empty string identifying a service in PortaBilling®:

- Session – Generic time-based service type; it can be used to apply charges for any service use based on the length of time the service was accessed.
- Voice – Rating telephony calls (incoming or outgoing) made via PortaSIP®, VoIP gateways or other equipment.
- Data – Data transfers rated using the amount transferred as the billing parameter.
- Netaccess – Internet access sessions (DSL, PPPoE, etc.), rated based on session duration or the amount of transferred data.
- Quantity – Generic quantity-based service type; can be used to apply charges for any service use expressible in numerical form (e.g. the number of pizzas ordered).
- Wifi – Wireless Internet access sessions, rated based on session duration.

- Dialup – Dialup Internet access sessions, rated based on session duration.
- Msg – Rating messages (text, SMS, MMS, other) based on the number of messages sent.
- Conference – Rating conference calls via PortaSIP® Media Server (or some conferencing server).
- Did – Can be used to apply charges to customers for usage of DID numbers.

Example:
```
PortaOne-Service-Type=Conference
```

PortaOne-Calling-Party

This provides information about the calling party.

Format:
```
PortaOne-Calling-Party = <propertyname1=value1>;…
```

Properties:

Property	Description
id	This contains the caller ID information for the call. It is taken from the *P-Asserted-Identity* and *RPID* headers
display-id	This indicates the “Caller number” value that is taken from the From: header and typically displayed on the called party’s phone display
privacy	This is a privacy flag which indicates that the calling party requests its identification to be kept private (hidden from the final call recipient.) Possible values: 1/0, default 0
name	This is a caller name. It is taken from the CLN field
x-id	The caller’s extension number configured on the PBX of a calling party
x-grp	The huntgroup extension, on behalf of which the call has happened
x	The unique identifier for the calling party PBX
media-time	The time the respective party spent connected to media on the call
bill-to	The account ID representing the entity in system. If not present, it means the party is from the outside network. The value can be used in User-Name if the respective party performs some action (e.g. becomes transferor).
net	The identifier of the network: • 0 for directly owned accounts. • >0 is an id of reseller.

Example:

```
h323-ivr-out     = 'PortaOne-Calling-Party =
'x=52333;name=EASYCALL;id=17289277770;display-id=17289277770''
```

PortaOne-Redirecting-Party

This provides information about the redirecting party in case the call is forwarded or transferred.

Format:
```
PortaOne-Redirecting-Party:<propertyname1=value1>;…
```

The properties and values are the same as for the *PortaOne-Calling-Party* attribute described above.

PortaOne-Media-Info

This provides information about the consumed bandwidth according to the formula reflecting the connectivity and media.

Format:
```
PortaOne-Media-Info:<propertyname1=value1>;…
```

Properties:

Property	Description
direction	Direction of the RTP stream on the leg relative to proxy
type	The mime type style description of the codec
ptime	The packetization time in milliseconds, which determines the amount of audio data included in a single packet
bitrate	The selected bitrate in bits per second
proxy	Indication of whether the RTP proxy was used. 0 - default

Example:
```
PortaOne-Media-Info =
'bitrate=64000;direction=in;type=audio/PCMU;proxy=1;ptime=20'
```

PortaOne-Geo-Location

This provides information about a user’s location.

Format:
`PortaOne-Geo-Location:`<the geo location specifier in dot notation starting with iso country code>

Example:
```
PortaOne-Geo-Location:fr.Paris
```

PortaOne-Service-Features

Format:

```
PortaOne-Service-Feature:<propertyname1=value1>;…
```

Example:

```
PortaOne-Service-Features = 'routing=1;'
```

PortaOne-Service-Action

This allows NAS to perform special actions when BE responds with the authorization.

Format:

```
PortaOne-Service-Action:action=<action type>;param1=<param1
value>;…
```

Example:

```
PortaOne-Service-Action = 'action=call-pickup;id=call-pickup;'
```

PortaOne-Charge

This informs that the imported xDR must be charged.

Format:

```
PortaOne-Charge:<propertyname1=value1>;…
```

Properties:

Property	Description
type	The type of the charged entity. Possible values: account/reseller/reseller[level]/vendor
storage-type	The type of the backend storage where this xDR record is added: Account/AccountFail, Customer/CustomerFail, Vendor/VendorFail, OnNet
charged-amount	The value of *xdr .charged_amount*
charged-quantity	The value of *xdr.charged_quantity* (in units/rating base according to the service)
i_account/i_customer/i_vendor	Identifier of the entity being charged
flag.*	The indicators associated with the charge. For example, flag.block=<threshold> informs that usage already passed into the discount stage, where the service is blocked and that it happened on a threshold with the given value

Example:

```
PortaOne-Charge = 'history=prerated;entity-
type=Account;charged-amount=1.10000;'
```

PortaOne-Called-Party

This provides information about the called party.

Format:
`PortaOne-called-Party:<propertyname1=value1>;…`

Properties:

Property	Description
id	This contains the caller ID information for the call. It is taken from the *P-Asserted-Identity* and *RPID* headers
display-id	This indicates the "Caller number" value that is taken from the From: header and typically displayed on the called party's phone display
privacy	This indicates that the calling party requests its identification to be kept private (hidden from the final call recipient.) Possible values: 1/0, default 0
realm	Optional. Realm for identifying the scope for called party identity
name	This is a caller name. It is taken from the CLN field
x-id	The caller's extension number configured on the PBX of a calling party
x	The unique identifier for the calling party PBX
x-grp	Optional. The group setting for the call group pickup. The huntgroup extension, on behalf of which the call happened
media-time	The time the respective party spent connected to media on the call
bill-to	The account ID representing the entity in system. If not present, it means the party is from the outside network. The value can be used in User-Name if the respective party performs some action (e.g. becomes transferor).
net	The identifier of the network: • 0 for directly owned accounts. • >0 is an id of reseller.

Example:

```
h323-ivr-out     = 'PortaOne-Calling-Party =
'x=52333;name=EASYCALL;id=17289277770;display-id=17289277770''
```

PortaOne-Transport-Protocol

This shows the transport protocol. Type can be one of the following:

- SIP
- H323
- SMS
- SMPP

Format:
```
PortaOne-Transport-Protocol = <type>
```

Example:
```
h323-ivr-out      ='PortaOne-Transport-Protocol = 'SIP''
```

PortaOne-Leg-Type

This shows whether the request is originate or answer. Type can be one of the following:

- 1 – Answer request
- 2 – Originate request

Format:
```
PortaOne-Leg-Type:<type>
```

Example:
```
h323-ivr-out      = PortaOne-Leg-Type = '1''
```

PortaOne-Leg-Origin

This shows the leg origin.

Format:
```
PortaOne-Leg-Origin:<type>
```

Example:
```
h323-ivr-out      = PortaOne-Leg-Origin = '0''
```

PortaOne-Used-Resource

This specifies the amount of used service units measured from the point when the service session identified in the h323_conf_id attribute becomes active.

Format:
```
PortaOne-Used-Resource:base=<base>;amount=<value>
```

Possible values:
<base> – One of: quantity, session-time
<value> – Integer number specifying amount of measurement units.

Example:
```
PortaOne-Used-Resource = base=quantity;value=2
PortaOne-Used-Resource = base=session-time;value=70
```

Attributes sent by the PortaBilling® server to a RADIUS client (gateway)

Attribute	A1	A2	Description
DURATION		Y	The maximum allowed call duration (in seconds)
Tariff	Y	Y	The name of the rate plan applied to the account within this session
available-funds	Y		For debit accounts, this is equal to the h323-credit-amount. For credit accounts, this returns the actual amount of available funds (the difference between the credit limit and the current balance with respect to the credit limits for the individual account and customer)
h323-preferred-lang	Y	Y	Language preference for the audio prompt
h323-credit-amount	Y		Amount of credit (in currency) in the account
h323-credit-time		Y	Amount of seconds for which the call is authorized
PortaBilling_UserName	Y	Y	The account ID to be used for billing this call
PortaBilling_RatePattern		Y	The rate pattern is a string used by PortaBilling® to match the corresponding rate code/destination during authorization of an account, instead of the Called-Station-Id
PortaBilling_AccessCode	Y	Y	Access code to be used for matching rating entry and tariff
PortaBilling_AccountBalance	Y		The current amount of the account's balance in PortaBilling®
PortaBilling_CustomerBalance	Y		The current amount of the customer's balance in

			PortaBilling®
PortaBilling_CustomerCreditLimit	Y		The customer's current credit limit in PortaBilling®
PortaBilling_CustomerCreditLimitThreshold	Y		The customer's current balance warning in PortaBilling®
BalanceThreshold	Y		Indicates that a customer's balance warning threshold has been reached
PortaBilling_ProductBreakage	Y		The breakage amount of the account's product in PortaBilling®
PortaBilling_No_Disconnect_Warning		Y	Indicates that NAS won't play a "time left" warning (the remaining time announcement in seconds or a "beeping" sound) when a specified number of seconds is left before the call is disconnected (calls will be terminated silently)

DURATION

This specifies the real allowed maximum call duration (in seconds), which may differ from the announced credit time if billing tricks are applied.

Format:

```
DURATION:<integer>
```

Possible values:
Positive integer (number of seconds).

Example:

```
h323-ivr-in       = 'DURATION:320'
```

Tariff

The name of the rate plan applied to the account within this session.

Format:

```
Tariff:<string>
```

Possible values:
Positive integer (number of seconds).

Example:
```
h323-ivr-in       = 'Tariff:ABC prepaid'
```

Available-funds

For debit accounts, this is equal to the h323-credit-amount. For credit accounts, this returns the actual amount of available funds (the difference between the credit limit and the current balance with respect to the credit limits for the individual account and customer).

Format:
```
available-funds:<decimal>
```

Possible values:
Any positive number formatted with two decimal places; a dot (.) is used as the decimal separator.

Example:
```
h323-ivr-in       = ' available-funds:124.78'
```

PortaBilling_UserName

The account ID to be used for billing this call. The RADIUS client **must** supply this value as the `User-Name` (or, alternatively, in `h323-ivr-out=PortaBilling_UserName`) attribute value in the accounting records for all call legs (incoming and outgoing).

Format:
```
PortaBilling_UserName:<value>
```

Possible values:
A non-empty string identifying an account in PortaBilling®.

Example:
```
h323-ivr-in       = 'PortaBilling_UserName:16051233355'
```

PortaBilling_RatePattern

The rate pattern is a string used by PortaBilling® to match the corresponding rate code/destination during authorization of an account, instead of the Called-Station-Id (e.g. when a customer is to be billed according to a special rate such as VOICEONNET, rather than according to the actual number dialed). The RADIUS client **must** supply this value as the `h323-ivr-out=PortaBilling_RatePattern` attribute value in the accounting records for all outgoing legs of this call.

Format:
```
PortaBilling_RatePattern :<value>
```

Possible values:

A non-empty string, usually a destination number in e.164 format.

Example:
```
h323-ivr-in      = 'PortaBilling_RatePattern:18001233355'
```

PortaBilling_AccessCode

The string actually used by PortaBilling® to match the corresponding accessibility entry and tariff. NAS should copy this value into all future call accounting requests as h323-ivr-out=PortaBilling_AccessCode, in order to apply the same properties during billing.

Format:
```
PortaBilling_AccessCode:<value>
```

Possible values:
A non-empty string, the same as for h323-ivr-out=PortaBilling_AccessCode.

Example:
```
h323-ivr-in      = 'PortaBilling_AccessCode:18001234567'
```

PortaBilling_AccountBalance

The current amount of the account's balance in PortaBilling®.

Format:
```
PortaBilling_AccountBalance:<value>
```

Possible values:
A number with a precision of five decimal places.

Example:
```
h323-ivr-in      = 'PortaBilling_AccountBalance:13.20000'
```

PortaBilling_CustomerBalance

The current amount of the customer's balance in PortaBilling®.

Format:
```
PortaBilling_CustomerBalance:<value>
```

Possible values:
A number with a precision of five decimal places.

Example:
```
h323-ivr-in      = 'PortaBilling_CurstomerBalance:13.20000'
```

PortaBilling_CustomerCreditLimit

The current amount of the customer's credit limit in PortaBilling®.

Format:
```
PortaBilling_CustomerCreditLimit:<value>
```

Possible values:
A number with a precision of five decimal places.

Example:
```
h323-ivr-in       = 'PortaBilling_CustomerCreditLimit:13.00000'
```

PortaBilling_CustomerCreditLimitThreshold

The current amount of the customer's credit limit threshold in PortaBilling®. Note that the relative threshold limit is converted into an actual currency amount.

Format:
```
PortaBilling_CustomerCreditLimitThreshold:<value>
```

Possible values:
A number with a precision of five decimal places.

Example:
```
h323-ivr-in       =
'PortaBilling_CustomerCreditLimitThreshold:12.00000'
```

BalanceThreshold

Indicates that the customer's balance warning threshold has been reached.

Format:
```
BalanceThreshold:<value>
```

Possible values:
```
    YES
```

Example:
```
h323-ivr-in       = 'BalanceThreshold:YES'
```

PortaBilling_ProductBreakage

The breakage amount of the account's product in PortaBilling®.

Format:
```
PortaBilling_ProductBreakage:<value>
```

Possible values:
A number with a precision of five decimal places.

Example:

```
h323-ivr-in      = 'PortaBilling_ProductBreakage:0.02000'
```

PortaBilling_No_Disconnect_Warning

Indicates that NAS should cancel IVR/beep warnings before call session timeouts (calls will be terminated silently).

Format:

```
PortaBilling_No_Disconnect_Warning:<value>
```

Possible values:

```
    YES
```

Example:

```
h323-ivr-in      = 'PortaBilling_No_Disconnect_Warning:YES'
```

RADIUS return codes

Return codes can be defined by the user through the implementation of the IVR script and RADIUS server agreement. The following table lists the return codes that are predefined and anticipated by existing Cisco IVR scripts.

Explanation code	h323-return-code	Description
success	0	Success, proceed
invalid_account	1	Failed - Invalid Account number
invalid_password	2	Failed - Invalid Password
account_in_use	3	Failed - Account in use
zero_balance	4	Failed - Zero balance
card_expired	5	Failed - Card expired
credit_limit	6	Failed - Credit limit
user_denied	7	Failed - User denied
not_available	8	Failed - Service not available
cld_blocked	9	Failed - Called number blocked
retries_exceeded	10	Failed - Number of retries exceeded
invalid_argument	11	Failed - Invalid argument
insuff_balance	12	Failed - Insufficient balance
toll_free_allowed	13	Toll-free calls allowed
invalid_card	14	Failed - Invalid card number
hairpin_to_pstn	50	Call will be hairpinned back to the Public Switched Telephone Network (PSTN)

redirect	51	Redirect to called party (use redirect number)
redirect_to_cs	52	Redirect to customer Service (use redirect number)

2. External System Provisioning Framework

External System Provisioning Framework (ESPF)

Billing, service provisioning, customer management and rating of services in PortaBilling® happens in real time, so that service configuration becomes effective as soon as it is updated. For example, when you create a new customer and new accounts – they can immediately start using services within PortaSwitch® once you press **Save**. When the customer uses a service (e.g. makes a call) – billing data (e.g. the customer's balance) is updated as soon as the session has completed.

But how do you provision other platforms to provide services like IPTV? The external system provisioning framework (ESPF) does this job. You configure the ESPF to capture specific events in the database (e.g. an account's balance has changed or the account has been blocked). When such an event occurs, a special event handler (a Perl module or even a bash script) sends it to the external system (e.g. removes the account in the IPTV platform if it was blocked in PortaBilling).

If an issue occurs during event provisioning, the corresponding error is written to the sokoban.log file. Besides, administrators can receive email notifications about provisioning event failures. On the Configuration server, they must specify an email address for the **ProvisioningFramework.AlertRecipient** parameter.

This allows administrators to intercept and resolve event-provisioning issues before they have any significant impact on end users.

How it works

When the administrator sets up the ESPF to catch a specific event, the following steps are implemented:

1. A database trigger is created for a particular table.
2. The database automatically executes the trigger in response to the defined event and the trigger inserts an event record into the Event_Queue table.
3. This record contains references to other tables with certain fields that can be provisioned to the external system (e.g. i_env, id and balance of an account).
4. A handler subscribed for this event reads the record from the table and provisions the event and its data to the external system and receives the return code (whether the event was processed successfully or not) from it.

5. If the return code is 1 (success), then ESPF removes the processed event from the Event_Queue table.

How to configure

To provision a specific event to an external system, follow these steps:

1. Enable the ESPF on the Configuration server web interface.
2. Find an available event handler to suit your needs (adjust the target URL and other required parameters) or create your own. Find event handlers that are supplied with PortaSwitch® **here**.
3. Enable the specified events to be triggered. Find the supported events **here**.
4. Enable the handler that will provide the information to the external system.
5. Subscribe the handler to catch the required events.

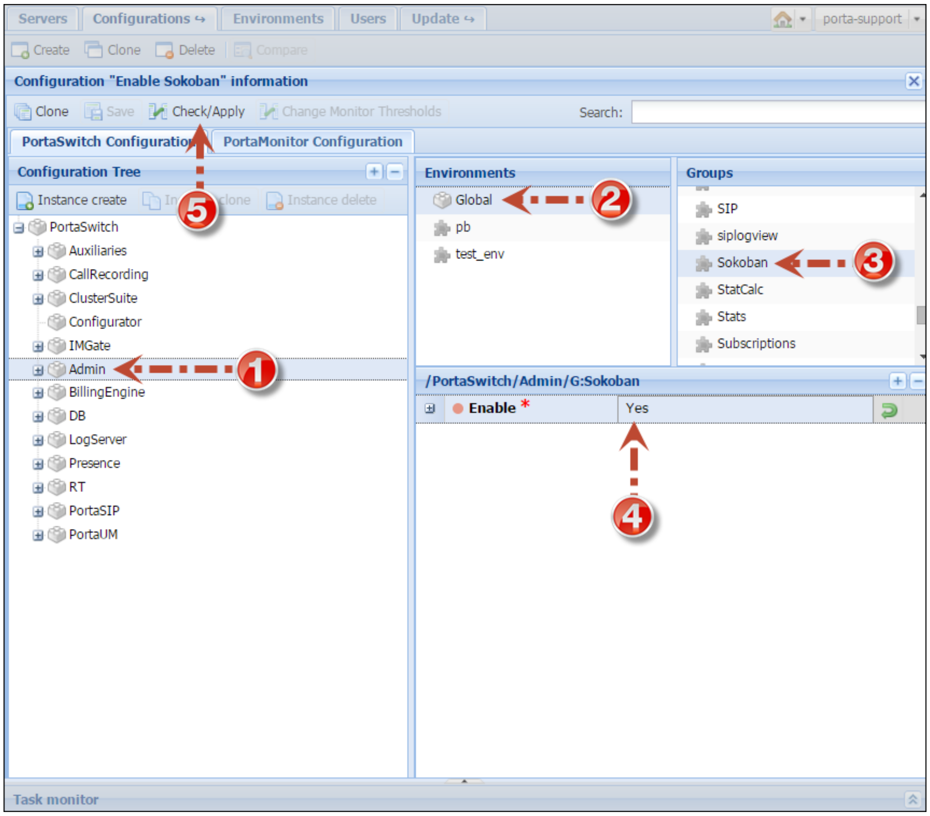
Example

Let's look at a simple example of how to configure the ESPF.
We send a POST request to the external system (find an example in the *An Example of a Request Processing Script* section) each time there's a change to an account balance in PortaBilling® (when either a service has been used or a payment made). The request will have the following information:

- The unique ID of the account database record (i_account);
- The current balance (current_balance);
- The previous balance (previous_balance).

Your configuration will likely differ from the one described in this example, but the general steps will be as follows:

Enable ESPF on the Configuration server



/PortaSwitch/Admin/G:Sokoban

Enable * Yes

Create your own event handler

Event handlers are located in the following folder:
`/home/provisioning-framework/site_lib/Porta/Event/Handler/`

We recommend that you modify DemoHandler.pm to your needs, since it is already registered in PortaBilling® and can be immediately used.

The event handler in our example will be as follows:

```
package Porta::Event::Handler::DemoHandler;

use strict;
use warnings;
use Data::Dumper;
use LWP::UserAgent;

use Porta::Event::Constants qw(:status);
use base qw(Porta::Event::Handler);

use Porta::TaskLog ();

sub process_one {
  my ($self, $queue, $notification) = @_;

  my $event_name = $notification->{event}->type_object->name;
  my $vars = $notification->{event}->variables;
  my $ua = LWP::UserAgent->new();
  if (exists $vars->{i_account}){
    my $response = $ua->post(
'https://api.example.com/webhook', {
        i_account => $vars->{i_account},
        previous_balance => $vars->{prev_balance},
        current_balance => $vars->{curr_balance},
        token => 'a13f7vna1dd4fblof4b1ood2knnfy3ndc1g',
      });
    Porta::TaskLog->debug('Webhook result: %s',
Dumper($response));
  }

  return OK;
}


1;
```

If you create your own handler from scratch (e.g. MyOwnHandler), add it to the folder with handlers:

```
/home/provisioning-framework/site_lib/Porta/Event/Handler/
```

Restart ESPF process

Each time you introduce changes to the event handler you must restart the ESPF process:

```
sudo service provisioning restart
```

Enable the trigger for the required event

```
/home/provisioning-framework/utils/evctl.pl type enable
Account/BalanceChanged
```

Enable the event handler

To enable your new handler, first insert the information about it and the event type into the database:

```
INSERT INTO Event_Subscriptions (i_event_type, i_event_handler,
description) VALUES (132,900, 'balance changed');
```

where `132` is the internal ID of the event type and `900` is the internal ID of your handler.

NOTE: Use the `Event_Types` table to find the ID of the event type and the `Event_Handlers` table to find the ID of your handler in the database.

Enable your handler:

```
/home/provisioning-framework/utils/evctl.pl handler enable
MyOwnHandler
```

To enable the modified DemoHandler, use the following command:

```
/home/provisioning-framework/utils/evctl.pl handler enable
DemoHandler
```

Subscribe the event handler for the event

```
/home/provisioning-framework/utils/evctl.pl handler sub
DemoHandler Account/BalanceChanged
```

Repeat this command to subscribe the handler for other events as well.

Make tests

Check that the required handler is enabled and subscribed for all specified events:

```
/home/provisioning-framework/utils/evctl.pl matrix
```

- "." shows that the event handler can subscribe for an event;
- "+" shows that the event handler is already subscribed for an event.

When an account's balance changes, you see this event in the log file:

```
/porta_var/<server_IP_address>/log/sokoban.log
```

Supplied event handlers

Event handlers are located in the following folder:

```
/home/provisioning-framework/site_lib/Porta/Event/Handler/
```

To use the following event handlers you need to adjust the target URL and other parameters and then restart the ESPF process:

- **AudioCodes.pm** – This provisions the AudioCodes CMMS (Clients' Management Server) which is responsible for customized profiles, user level control, time-based licensing and other functions.
- **AutoSettingOfCreditLimitAndVD.pm** – This automatically sets a customer's credit limit and volume discount plan.
- **Bridgewater.pm** – This provisions Bridgewater, the service controller for WiMAX networks.
- **DemoHandler.pm** – This is an example of the event handler. Use it to create a custom handler.
- **EPC.pm** – This is used for LTE service provisioning. It provisions customer data to Greenisis HSS (for centralized user authentication across the network) and PCRF (to allocate the correct policy / QoS parameters to the customer).
- **LTE.pm** – This is used for LTE service provisioning. It provisions customer data to Huawei HSS (for centralized user authentication across the network) and PCRF (to allocate the correct policy / QoS parameters to the customer).
- **IPTV.pm** – This provisions available channel information to the IPTV platform when a new or existing customer signs up for the IPTV service.

- **NBBS.pm** – This provisions the Motorola NBBS Device Management Platform.
- **NetNumber** – This provisions ported-in numbers to the NetNumber SMS/MMS database.
- **ProductToAccountsDispatcher.pm** – This processes all changes on the product level and provisions them to an account with this product assigned.
- **Protei.pm** – This is used for provisioning subscriber information and service configuration data to Protei HSS (for centralized user authentication across the network) and PCRF (to allocate the correct policy / QoS parameters to the customer).
- **ReadyWireless.pm** – This provisions account data to ReadyWireless MVNE to trigger device activation / deactivation.
- **ServiceAttributeDispatcher.pm** – This provisions changes in service configuration to customers and accounts. You can specify filters for this event handler to reduct the number of quieries to the database. See the **Filters** section for details.
- **SIPForwarder** – This is used when an account is moved from one system to another. It provisions information about an account and / or DID number location to the dispatching SBC in the dual-version PortaSwitch® deployment.
- **ZTEEPC.pm** – This provisions the LTE / 3G services to the ZTE EPC and supports quota control for these services.

Events that can be triggered

The following command shows which supplied event handlers can trigger specific events:

```
/home/provisioning-framework/utils/evctl.pl matrix
```

The following table shows all the events that can be triggered by ESPF:

Name	Description	Inherited from Customer
Account / New	A new account has been created	
Account / Status / Changed	A customer's status has changed (Customers.bill_status has changed)	Y
Account / Status / Suspend	A customer's account has been suspended (Customers.bill_status set to 'S')	Y

Account / Status / Unsuspend	A customer's account has been reinstated (Customers.bill_status set to 'O' from 'S')	Y
Account / Status / Closed	An account has been closed (Accounts.bill_status set to 'C')	Y
Account / Blocked	An account has been blocked (Accounts.blocked set to 'Y')	
Account / Unblocked	An account has been unblocked (Accounts.blocked set to 'N')	
Account / Product / Changed	An account's product has changed (Account.i_product has been updated)	
Account / ServiceFlags / Changed	An account's service features have changed (Accounts.service_flags has been updated)	Y
Account / UMEnabled / Changed	An account's UM feature has changed (Accounts.um_enabled has been updated)	
Account / FollowMe / Changed	An account's follow-me setting has changed (the Follow_Me table has been updated)	
Account / ServiceAttribute / Changed	An account's service setting has changed (the Service_Attribute_Values table has been updated)	Y
Customer / New	A new customer has been created	
Customer / Status / Changed	A customer's status has changed (Customers.bill_status has been updated)	
Customer / Status / Suspend	A customer's account has been suspended (Customers.bill_status has been updated)	

Customer / Status / Unsuspend	A customer's account has been reinstated (Customers.bill_status set to 'O' from 'S')	
Customer / Status / Closed	A customer's account has been closed (Customers.bill_status set to 'C')	
Customer / Status / Deactivated	A customer's account has been deactivated (Customers.bill_status set to 'D' from 'O')	Y
Customer / Status / Activated	A customer's account has been activated (Customers.bill_status set to 'O' from 'D)'	Y
Customer / Blocked	A customer's account has been blocked (Customers.blocked set to 'Y')	Y
Customer / Unblocked	A customer's account has been unblocked (Customers.blocked set to 'N')	Y
Customer /ServiceFlags / Changed	A customer's service features have changed (Customers.service_flags has been updated)	
Customer / ServiceAttribute / Changed	A customer's service setting has changed (the Service_Attribute_Values table has been updated)	
ServiceAttribute / Changed	The Service_Attribute_Values table has been updated	
Account / IPDeviceAssignment	An IP device has been assigned to an account (the UA_Links table has been updated)	
IPDeviceProfile / New	Created new UA_Profiles	
CustomField / Changed	A custom field has changed (the Custom_Field_Values table has been updated)	

Customer / CustomField / Changed	A customer’s custom field has changed (the Custom_Field_Values table has been updated for the customer)	
Account / CustomField / Changed	An account’s custom field has changed (the Custom_Field_Values table has been updated for the account)	
Account / ZeroAvailableFunds	An account’s available funds have been depleted (Accounts.balance has reached credit_limit (for credit accounts) or 0 (for debit accounts))	Y
Customer / ZeroAvailableFunds	A customer’s available funds have been depleted (Customers.balance has reached credit_limit)	
Account / AvailableFundsAppear	An account’s available funds have been restored (Accounts.balance has been topped up from credit_limit)	Y
Customer / AvailableFundsAppear	A customer’s available funds have been restored (Customers.balance has been topped up from credit_limit)	
Account / Id / Changed	An account’s ID has been changed (Account.id has been updated)	
Accessibility / Inserted	A product’s rating entry has been added	
Accessibility / Deleted	A product’s rating entry has been removed	
Product / ServiceAttribute / Changed	Service setting has changed on a product (the Service_Attribute_Values table has been updated)	
Account / ActivationDate / Changed	An account’s activation date has changed (Accounts.activation_date has been changed)	

Account / ExpirationDate / Changed	An account’s expiration date has changed (Accounts.expiration_date has been changed)	
Account / Password / Changed	An account’s password for the self-care interface has changed (Accounts.password has been changed)	
Subscriber / Name / Changed	The name of an account’s user has changed	
Subscriber / Address / Changed	The address of an account’s user has changed	
Account / FollowMeNumber / Inserted	A new follow-me number has been added for an account	
Account / FollowMeNumber / Changed	A follow-me number has been changed for an account	
Account / FollowMeNumber / Deleted	A follow-me number has been removed from an account	
Account / ProductAddon / Inserted	A new add-on product has been added for an account	
Account / ProductAddon / Deleted	An add-on product has been removed from an account	
Account / ProductAddon / Changed	An add-on product has been changed for an account	
Account / ServicePassword / Changed	An account’s service password has changed (Accounts.h323_password has been changed)	
Account / Alias / Delete	An account’s alias has been removed	
Customer / Name / Changed	A customer’s name has changed	
Account / Status / Exported	A customer’s account status has changed (Customers.bill_status set to E)	Y

Account / Status / Imported	A customer's account status has changed (Customers.bill_status set to O)	Y
Account / Service / QuotaExceeded	An account has exceeded its service usage quota	
Account / Service / QuotaAvailable	The service usage quota has been restored for an account (quota counter has been reset or user topped up their service wallet)	
Subscriber / ContactInfo / Changed	The contact information has been changed for an account	
Account / SIMCardAssignment	An account has been assigned a SIM card from the SIM card inventory	
Account / Discount / Changed	A discount plan has been changed for an account	
Product / Discount / Changed	A discount plan has been changed within a product configuration	
DID / New	A new DID number has been added	
DID / Status / Assigned	A DID number has been assigned to a customer	
DID / Status / Unassigned	A DID number has been released from a customer	
DID / Status / Activated	A DID number has been assigned to a customer's account ID	
DID / Status / Canceled	A DID number has been unassigned from a customer's account ID	
DID / Status / Moved	A DID number has been moved to another installation as a result of provisioning	
DID / Deleted	A DID number has been removed from the DID inventory	
Customer / Status / Exported	A customer's account status has changed (Customers.bill_status set to E)	

Customer / Status / Imported	A customer's account status has changed (Customers.bill_status set to O)	
Invoice/New	A new invoice has been generated.	
Invoice/Status/Changed	An invoice has changed its status from: - unpaid to paid; - unpaid to partially paid; - partially paid to paid in full. Note that if a customer pays their invoice in installments, the Invoice/Status/Changed event occurs twice for this invoice: - first event – changes from unpaid to partially paid; - second – changes from partially paid to paid in full. E.g., A customer pays their $110 invoice in installments (e.g. $25+$20+$35+$30). Events occur when: - the first $25 payment is made: the invoice status changes from unpaid to partially paid; the last $30 payment is made: the invoice status changes from partially paid to paid in full.	

Invoice/AmountPaid/ Changed	An amount a customer has paid. Use this event to track and provision: - manual payment information; partially paid invoice changes.	
Invoice/Adjustment/ Changed	An invoice has been adjusted (refunded) by an administrator.	
PaymentTransaction/New	A customer made a credit card payment.	
PaymentTransaction/ ResultCode/Changed	A transaction has either started / finished.	

Filters

The ServiceAttributeDispatcher event handler generates provisioning events with each change in account or product service configuration. For some services (e.g. LTE), it must only provision changes related to that service. Thus, if an account uses Internet access and an LTE service, provisioning events must be triggered only when the LTE service configuration changes.

The flexible filtering mechanism does just that. When enabled for the ServiceAttributeDispatcher event handler, it only tracks the changes of particular service attributes and triggers provisioning events. This reduces the number of queries to the database and speeds up provisioning.

Supported filters

lte

This filter is applicable to LTE-related handlers: LTE, EPC, ZTEEPC, Protei.

When enabled, it tracks only the changes in the Access Policy and Static IP service features and their respective attributes: Internet Access Policy name, Static IP, PDP / PDN Context ID. Changes within all other service features and attributes are ignored.

Enable this filter only if you do not need to provision other service attribute changes (e.g. for IPTV services, etc.) and have not enabled other handlers that subscribe to them.

To enable a filter, execute the following command:

```
/home/provisioning-framework/utils/evctl.pl handler filter
ServiceAttributeDispatcher lte
```

To disable a filter, execute the following command:

```
/home/provisioning-framework/utils/evctl.pl handler filter
ServiceAttributeDispatcher none
```

As a result, the filter name is displayed right after the handler name:

```
/home/provisioning-framework/utils/evctl.pl handler list
enabled
```

```
ID   Handler                         Enabled   Description
3    ServiceAttributeDispatcher|lte  Yes       Dispatch events
                                               to Accounts /
                                               Customers
4    DemoHandler                     Yes       Example
106  LTE                             Yes       Huawei LTE
                                               provisioning
```

Configuration particularities

Consider the following points when configuring the ESPF:

- Enabling triggers on events causes additional load to the database;
- Each triggered event can be processed by more than one event handler. In this case, the event will be removed from the Event_Queue table only after the last handler processes it.
- To subscribe an event handler to all events that it supports execute the following command:
  ```
  /home/provisioning-framework/utils/evctl.pl handler
  sub <event_handler_name>
  ```
- To see the ESPF help, execute the following command:
  ```
  /home/provisioning-framework/utils/evctl.pl --help
  ```
- To ensure that your custom modifications remain on your servers during a software upgrade, list them in the Deposits page on the Configuration server web interface.

An example of a request processing script

Below is a 7-line script that was created using Flask (a python microframework). Execute the following command in Linux to install Flask using pip (a package management system used to install and manage software packages written in Python):

```
sudo pip install flask
```

The script receives requests with data from a handler and writes the data into the log file:

```
from flask import Flask, request
app = Flask(__name__)
@app.route('/test', methods=['POST', 'GET'])
def test():
    print(request.get_data())
    return "<h1>OK</h1>"
app.run(debug=True, port=3000, host="0.0.0.0")
```

3. Appendices

Examples of RADIUS communication

Failed login (Authentication failure)

Authentication request

```
NAS-IP-Address = '193.38.81.226'
User-Name  = '11380441234567'
Calling-Station-Id = '11380441234567'
Service-Type = '15'
h323-conf-id = 'BEEF0960 159C48A3 5CA4966B 7536B52C'
call-id = '94fef899-3f606c5c@192.168.192.59'
Digest-Username = '11380441234567'
Digest-Realm = '193.38.81.226'
Digest-Nonce = '1373028145:4ed5168d6120ae148bc32520fdf6bba2'
Digest-URI = 'sip:193.38.81.226'
Digest-Method = 'REGISTER'
Digest-Algorithm = 'MD5'
Digest-Response = '1984af4a0e13ceb54ae86e015e934bc5'
h323-remote-address = '192.168.192.59'
NAS-Port = '5060'
```

Authentication reject response

```
Reply-Message = Failed - Invalid Account number
h323-return-code = 1
h323-ivr-in = ErrorExplanation:invalid_account
```

Failed outgoing call (Authorization failure)

Authorization request

```
NAS-IP-Address = '193.38.81.226'
User-Name = '380441234567'
Called-Station-Id = '91245684289'
Calling-Station-Id = '380441234567'
h323-conf-id = 'BC0050CE E4B011E2 B062000C 29E9476D'
call-id  = 'f7442be6-3c18ddd@192.168.192.59'
Digest-Username = '380441234567'
Digest-Realm = '193.38.81.226'
Digest-Nonce = '51bec9bb7e2276edaf48b8572a3f72300ec0'
Digest-URI = 'sip:91245684289@193.38.81.226'
Digest-Method = 'INVITE'
Digest-Algorithm = 'MD5'
Digest-Response = '35139c9d1c67dd3192a02158d93eea50'
h323-remote-address = '192.168.192.59'
h323-session-protocol = 'sipv2'
h323-ivr-out = 'PortaBilling_AuthMethod:INVITE'
PortaOne-Service-Features = 'routing=1;version=25.0'
h323-ivr-out = 'PortaBilling_Seed:2912511541'
PortaOne-Calling-Party = 'name=380441234567'
h323-ivr-out =
'PortaSIP_Aux_Info:auth_life=134455/1/0;auth_cnt=1;CC_now=13445
5/1;pid=17785;request_type=routing'
NAS-Port  = '5060'
```

Authorization reject response

```
Reply-Message = Failed - Called number blocked
h323-return-code = 9
h323-ivr-in = ErrorExplanation:cld_blocked
h323-ivr-in = Product:EasyCall Residential
h323-ivr-in = Tariff:EasyCall Residential
h323-ivr-in = PortaBilling_NodeId:193.38.81.226
h323-ivr-in = PortaBilling_AccessCode:OUTGOING
h323-ivr-in = PortaBilling_Seed:1599467173
h323-billing-model = 0
h323-currency = USD
h323-preferred-lang = en
```

Successful prepaid card IVR session

Authentication request

```
NAS-IP-Address         = '164.9.9.100'
NAS-Port-Type          = 'Async'
User-Name              = '10086610975'
Calling-Station-Id     = '6045550193'
Service-Type           = 'Login-User'
h323-conf-id           = '39AE126B CD4D11DB 958E0014 1C3F6886'
Password               = ''
h323-ivr-out           = 'PortaBilling_AccessCode:6045551600'
h323-ivr-out           = 'transactionID:526267'
NAS-Port               = '0'
NAS-Port-Id            = 'ISDN 3/0:D:13'
```

Authentication response

```
h323-credit-amount     = 10.00
h323-billing-model     = 1
h323-ivr-in            = Tariff:PrepaidCard
h323-ivr-in            = PortaBilling_AccountBalance:10.00000
h323-ivr-in            = PortaBilling_ProductBreakage:0.00000
h323-ivr-in            = available-funds:10.00
h323-return-code       = 0
h323-currency          = CAD
h323-preferred-lang    = en
```

Authorization request

```
NAS-IP-Address         = '164.9.9.100'
User-Name              = '10086610975'
Called-Station-Id      = '01182623634515'
Calling-Station-Id     = '6045550193'
Service-Type           = 'Login-User'
h323-conf-id           = '39AE126B CD4D11DB 958E0014 1C3F6886'
Password               = ''
h323-ivr-out           = 'PortaBilling_Seed:18879071672'
h323-ivr-out           = 'PortaBilling_Original_CLD:6045551600'
h323-ivr-out           = 'transactionID:526273'
```

Authorization response

```
h323-billing-model     = 1
h323-ivr-in            = Tariff:PrepaidCard
h323-ivr-in            = PortaBilling_CompleteNumber:82623634515
h323-ivr-in            = PortaBilling_Auth_CLD:82623634515
h323-ivr-in            = PortaBilling_Auth_Reseller_CLD:82623634515
h323-ivr-in            = DURATION:30000
h323-return-code       = 0
```

```
h323-currency          = CAD
h323-credit-time       = 30000
h323-preferred-lang    = en
```

Accounting request (outgoing call leg)

```
NAS-IP-Address          = '164.9.9.100'
User-Name               = '10086610975'
Called-Station-Id       = '82623634515'
Calling-Station-Id      = '6045550193'
Acct-Status-Type        = 'Stop'
Service-Type            = 'Login-User'
h323-call-origin        = 'originate'
h323-call-type          = 'VoIP'
h323-setup-time         = '00:16:18.192 PST Fri Mar 9 2007'
h323-connect-time       = '00:16:21.164 PST Fri Mar 9 2007'
h323-disconnect-time    = '00:17:31.893 PST Fri Mar 9 2007'
h323-disconnect-cause   = '10'
h323-conf-id            = '39AE126B CD4D11DB 958E0014 1C3F6886'
call-id                 = '4A65E46C-CD4D11DB-886EDEBF-
7AF0CBAB@164.9.9.100'
h323-incoming-conf-id   = '39AE126B CD4D11DB 958E0014 1C3F6886'
h323-ivr-out            = 'PortaBilling_Original_CLD:6045551600'
h323-ivr-out            = 'PortaBilling_Seed:18879071672'
h323-remote-address     = '164.9.9.101'
remote-media-address    = '164.9.9.101'
```

Accounting request (incoming call leg)

```
NAS-IP-Address          = '164.9.9.100'
NAS-Port-Type           = 'Async'
User-Name               = '10086610975'
Called-Station-Id       = '6045551600'
Calling-Station-Id      = '6045550193'
Acct-Status-Type        = 'Stop'
Service-Type            = 'Login-User'
h323-call-origin        = 'answer'
h323-call-type          = 'Telephony'
h323-setup-time         = '00:15:50.148 PST Fri Mar 9 2007'
h323-connect-time       = '00:15:50.156 PST Fri Mar 9 2007'
h323-disconnect-time    = '00:17:31.889 PST Fri Mar 9 2007'
h323-disconnect-cause   = '10'
h323-conf-id            = '39AE126B CD4D11DB 958E0014 1C3F6886'
h323-incoming-conf-id   = '39AE126B CD4D11DB 958E0014 1C3F6886'
h323-ivr-out            = 'PortaBilling_Session:unlock'
h323-ivr-out            = 'Tariff:Unknown'
NAS-Port                = '0'
NAS-Port-Id             = 'ISDN 3/0:D:13'
```

SIP routing request

Authorization request

```
NAS-IP-Address = '193.28.87.217'
User-Name = '193.38.81.226'
Called-Station-Id = '12125468754'
Calling-Station-Id = '380441234567'
h323-conf-id = 'FD0158BA E2F911E2 B062000C 29E9476D'
call-id = '482bb2cb-aa0eada5@192.168.192.59~1o'
Password = 'cisco'
h323-remote-address = '193.38.81.226'
h323-session-protocol = 'sipv2'
h323-ivr-out = 'PortaBilling_AuthMethod:INVITE'
PortaOne-Service-Features = 'routing=1;version=25.0'
h323-ivr-out = 'PortaBilling_Seed:2567011031'
PortaOne-Calling-Party = 'id=380441234567;name=11380441234567'
```

```
h323-ivr-out = 'PortaBilling_Ignore_Password:YES'
h323-ivr-out =
'PortaSIP_Aux_Info:auth_life=164498/1/0;auth_cnt=1;CC_now=16449
8/1;pid=19218;request_type=routing'
NAS-Port = '5060'
```

Authorization response

```
h323-return-code = 13
h323-ivr-in = PortaBilling_CompleteNumber:12125468754
h323-ivr-in = Product:PortaOne Internal
h323-ivr-in = Tariff:Internal
h323-ivr-in = PortaBilling_NodeId:193.38.81.217
h323-ivr-in = PortaBilling_AccessCode:OUTGOING
h323-ivr-in = PortaBilling_Seed:2567011031
h323-ivr-in = PortaBilling_RatePattern:12125468754
h323-ivr-in = PortaBilling_Authorize:base=session-time;avail=-
1;expires=1373028131
h323-billing-model = 0
PortaOne-Service-Features = moh=1;rec=2;rtpp-level=3
PortaOne-Calling-Party =
name=11380441234567;id=380441234567;display-id=380441234567
h323-currency = USD
h323-preferred-lang = en
Framed-Route = r=;g-hunt=seq
Framed-Route = P_=;g-hunt=skip;expires=300;patience=20
Framed-Route =
p__=65.174.119.133;auth=AA14CF3595B86BF7B1C7B0458626C288DD5B419
1C0B759A3
Framed-Route = p__=69.25.184.13;rtpp-level=3
```

Callback session

First call leg authorization

```
NAS-IP-Address         = "142.179.71.220"
User-Name              = "00099900113"
Called-Station-Id      = "00099900113"
Calling-Station-Id     = "00099900113"
h323-conf-id           = "2F01F3F6 B89192B8 711B0000 1C7A0F2A "
Password               = "secret"
h323-remote-address    = "193.28.87.193"
h323-ivr-out           = "PortaBilling_OriginalCLD:555333"
h323-ivr-out           = "PortaBilling_AuthMethod:INVITE"
h323-ivr-out           = "PortaBilling_Seed:4094388346"
Message-Authenticator = '84C47B0BDC12887A60D2C5167C740A'
```

Second call leg authorization

```
NAS-IP-Address         = "142.179.71.220"
User-Name              = "00099900113"
Called-Station-Id      = "00099900111"
Calling-Station-Id     = "00099900222"
h323-conf-id           = "2F01F3F6 B89192B8 711B0000 1C7A0F2A"
Password               = "secret"
h323-ivr-out           = "PortaBilling_OriginalCLD:OUT"
h323-ivr-out           = "PortaBilling_AuthMethod:INVITE"
h323-ivr-out           =
PortaBilling_Notify:another_cld=00099900113;another_orig_cld=55
5333;in_progess=13"
h323-ivr-out           = "PortaBilling_Seed:4094388346"
```

```
Message-Authenticator = '12C47B0B16412247A60D2C5167C718C'
```

Second call leg 1, accounting for failed call attempt

```
NAS-IP-Address          = 142.179.71.220
h323-call-origin        = "originate"
h323-call-type          = "VoIP"
h323-setup-time         = "11:34:20.677 GMT Fri Feb 23 2007"
User-Name               = "00099900113"
Calling-Station-Id      = "00099900113"
Called-Station-Id       = "000999002222"
h323-incoming-conf-id   = "2F01F3F6 B89192B8 711B0000 1C7A0F2A"
h323-conf-id            = "11111111 B89192B8 711B0000 1C7A0F2A"
Acct-Session-Id         = "B89192B8-66BC43D89"
h323-disconnect-time    = "11:34:30.677 GMT Fri Feb 23 2007"
h323-connect-time       = "11:34:30.677 GMT Fri Feb 23 2007"
h323-ivr-out            = “PortaBilling_Seed:4094388346”
h323-remote-address     = "10.28.87.193"
h323-ivr-out            = "PortaBilling_OriginalCLD:OUT"
Acct-Session-Time       = 0
h323-disconnect-cause   = "1C"
Acct-Status-Type        = Stop
```

Second call leg, accounting for connected call

```
NAS-IP-Address          = 142.179.71.220
h323-call-origin        = "originate"
h323-call-type          = "VoIP"
h323-setup-time         = "11:34:35.677 GMT Fri Feb 23 2007"
User-Name               = "00099900113"
Calling-Station-Id      = "00099900113"
Called-Station-Id       = "00099900222"
h323-incoming-conf-id   = "2F01F3F6 B89192B8 711B0000 1C7A0F2A"
h323-conf-id            = "11111111 B89192B8 711B0000 1C7A0F2A"
Acct-Session-Id         = "B89192B8-66BC43D89"
h323-disconnect-time    = "11:34:30.677 GMT Fri Feb 23 2007"
h323-connect-time       = "11:36:32.677 GMT Fri Feb 23 2007"
h323-ivr-out            = "PortaBilling_Seed:4094388346"
h323-remote-address     = "10.28.87.200"
h323-ivr-out            = "PortaBilling_OriginalCLD:OUT"
Acct-Session-Time       = 122
h323-disconnect-cause   = "0"
Acct-Status-Type        = Stop
```

First call leg accounting

```
NAS-IP-Address          = 142.179.71.220
h323-call-origin        = "originate"
h323-call-type          = "VoIP"
h323-setup-time         = "11:33:38.677 GMT Fri Feb 23 2007"
User-Name               = "00099900113"
Calling-Station-Id      = "00099900113"
Called-Station-Id       = "00099900113"
h323-ivr-out            = PortaBilling_CallbackHistory:START
h323-ivr-out            = PortaBilling_CallbackHistory:ATTEMPT
h323-ivr-out            = PortaBilling_CallbackHistory:ATTEMPT
h323-ivr-out            = PortaBilling_CallbackHistory:OK
h323-conf-id            = "2F01F3F6 B89192B8 711B0000 1C7A0F2A"
Acct-Session-Id         = "B89192B8-66BC43D89"
h323-disconnect-time    = "11:34:18.677 GMT Fri Feb 23 2007"
h323-connect-time       = "11:35:18.677 GMT Fri Feb 23 2007"
h323-ivr-out            = "PortaBilling_Seed:4094388346"
h323-remote-address     = "10.28.87.200"
h323-ivr-out            = "PortaBilling_OriginalCLD:555333"
Acct-Session-Time       = 60
h323-disconnect-cause   = "0"
Acct-Status-Type        = Stop
Acct-Delay-Time         = 0
```

Voucher recharge session

Authentication (main account login) request

```
NAS-IP-Address          = '127.0.0.102'
User-Name               = '02001'
Calling-Station-Id      = '02001'
h323-conf-id            = '01010101 00000000 00000000 00000012'
PortaOne-Service-Type = 'Quantity'
h323-ivr-out            = 'PortaBilling_Original_CLD:01inbound'
h323-ivr-out = 'PortaBilling_Ignore_Password:YES'
Message-Authenticator = '52C47B0B16412147A60D2C5167C718D'
```

Authentication response

```
h323-credit-amount      = 10.00
h323-billing-model      = 1
h323-ivr-in             = PortaBilling_AccessCode:01inbound
h323-ivr-in             = Tariff:T_SubService_1
h323-ivr-in             = PortaBilling_AccountBalance:10.00000
h323-ivr-in             = PortaBilling_ProductBreakage:0.00000
h323-ivr-in             = available-funds:10.00
h323-return-code        = 0
h323-currency           = USD
h323-preferred-lang     = en
Message-Authenticator = '1CD6D5B0BDC33987A60D5C6167B22517'
```

Voucher recharge request

```
NAS-IP-Address          = '127.0.0.102'
User-Name               = '02001'
Calling-Station-Id      = '02001'
h323-conf-id            = '01010101 00000000 00000000 00000012'
h323-conf-id = 'PortaBilling_Session:ignore'
h323-ivr-out =
'PortaBilling_Mod_ivrutil:action=rechargeVoucher;id=111222336'
h323-ivr-out            = 'PortaBilling_Original_CLD:01inbound'
h323-ivr-out = 'PortaBilling_Ignore_Password:YES’
```

Voucher recharge response

```
h323-ivr-in             = PortaBilling_AccessCode:01inbound
h323-ivr-in =
'PortaBilling_Mod_ivrutil:action=rechargeVoucher;amount=50.00000'
h323-ivr-in             = PortaBilling_RechargedAmount:50.00000
h323-ivr-in             = Tariff:T_SubService_1
h323-ivr-in             = available-funds:60.00
h323-return-code        = 0
h323-credit-amount      = 60.00
h323-currency           = USD
h323-preferred-lang     = en
```

Music download service requests

Authorization request

```
NAS-IP-Address              = '192.168.100.50'
User-Name                   = '0101'
Password                    = 'secret'
Called-Station-Id           = 'JAZZ'
h323-conf-id                = 'A5D3B74C D26411DC 98D5001D 090C82FA'
PortaOne-Service-Type       = Quantity
```

Authorization response (for credit account without credit limit)

```
h323-billing-model   = 0
h323-ivr-in          = Tariff:Music Account
h323-ivr-in          = PortaBilling_CompleteNumber:JAZZ
h323-ivr-in          = PortaBilling_Auth_CLD:JAZZ
h323-return-code     = 13
h323-currency        = USD
h323-preferred-lang = en
Message-Authenticator = '1D86D5B0BDC33987A60D5C6167B13D91'
```

Authorization response (for debit account)

```
h323-billing-model   = 1
h323-ivr-in          = Tariff:Music Account
h323-ivr-in          = DURATION:10
h323-ivr-in          = PortaBilling_CompleteNumber:JAZZ
h323-ivr-in          = PortaBilling_Auth_CLD:JAZZ
h323-return-code     = 0
h323-currency        = USD
h323-credit-time     = 10
h323-preferred-lang = en
```

Accounting request

```
NAS-IP-Address         = '192.168.100.50'
User-Name              = '0101'
Called-Station-Id      = 'JAZZ'
Acct-Status-Type       = 'Stop'
h323-call-origin       = 'originate'
h323-call-type         = 'Telephony'
h323-setup-time        = '07:18:35.965 GMT Mon Feb 6 2008'
h323-connect-time      = '07:18:43.965 GMT Mon Feb 6 2008'
h323-disconnect-time   = '08:11:38.965 GMT Mon Feb 6 2008'
h323-disconnect-cause = '0'
h323-conf-id           = 'A5D3B74C D26411DC 98D5001D 090C82FA'
Acct-Session-Id        = 'da6b76'
Acct-Session-Time      = '3175'
PortaOne-Service-Type = Quantity
h323-ivr-out           = 'Used-Service-Unit:1'
```

Internet session

Authorization request

```
NAS-IP-Address                    = '41.213.1.32'
NAS-Port-Type                     = 'Ethernet'
User-Name                         = 'GP4899@ibhighspeed.co.za'
Called-Station-Id                 = 'pppoe-service1'
Calling-Station-Id                = 'E4:8D:8C:EE:1A:26'
Service-Type                      = 'Framed-User'
Mikrotik-Realm                    = 'ibhighspeed.co.za'
Unknown-Attribute                 = 'unknown vendor: attr
{311, 10, 0, 17, "ibhighspeed.co.za"}'
Framed-Protocol                   = 'PPP'
Event-Timestamp                   = '1514299872'
NAS-Identifier                    = 'WDSL_PPPoE2_MDC'
Password                          = 'password1'
NAS-Port                          = '15760440'
NAS-Port-Id                       = 'vlan140'
```

Authorization response

```
Service-Type                        = 'Framed-User'
Session-Timeout                     = '0'
Ascend-Data-Rate                    = '20000000'
Idle-Timeout                        = '86400'
Framed-IP-Address                   = '41.213.82.27'
Ascend-Xmit-Rate                    = '20000000'
Framed-IP-Netmask                   = '255.255.255.255'
Framed-MTU                          = '1500'
```

Start Accounting request

```
NAS-IP-Address                      = '41.213.1.32'
NAS-Port-Type                       = 'Ethernet'
User-Name                           = 'GP4899@ibhighspeed.co.za'
Called-Station-Id                   = 'pppoe-service1'
Calling-Station-Id                  = 'E4:8D:8C:EE:1A:26'
Acct-Status-Type                    = 'Start'
Service-Type                        = 'Framed-User'
Acct-Session-Id                     = '81707b87'
Acct-Delay-Time                     = '0'
Mikrotik-Realm                      = 'ibhighspeed.co.za'
Unknown-Attribute                   = 'unknown vendor: attr
{311, 10, 0, 17, "ibhighspeed.co.za"}'
Acct-Authentic                      = 'RADIUS'
NAS-Port-Id                         = 'vlan140'
Framed-Protocol                     = 'PPP'
NAS-Identifier                      = 'WDSL_PPPoE2_MDC'
Event-Timestamp                     = '1514299872'
Framed-IP-Address                   = '41.213.82.27'
NAS-Port                            = '15760440'
```

Keep-Alive Accounting request

```
NAS-IP-Address                      = '41.213.1.32'
NAS-Port-Type                       = 'Ethernet'
User-Name                           = 'GP4899@ibhighspeed.co.za'
Called-Station-Id                   = 'pppoe-service1'
Calling-Station-Id                  = 'E4:8D:8C:EE:1A:26'
Acct-Status-Type                    = 'Alive'
Service-Type                        = 'Framed-User'
Acct-Session-Id                     = '81707b87'
Acct-Input-Octets                   = '1013118'
Acct-Output-Octets                  = '599107'
Acct-Input-Packets                  = '3494'
Acct-Output-Packets                 = '3082'
Acct-Session-Time                   = '1800'
Acct-Delay-Time                     = '0'
Mikrotik-Realm                      = 'ibhighspeed.co.za'
Unknown-Attribute                   = 'unknown vendor: attr
{311, 10, 0, 17, "ibhighspeed.co.za"}'
Idle-Timeout                        = '86400'
NAS-Identifier                      = 'WDSL_PPPoE2_MDC'
Framed-IP-Address                   = '41.213.82.27'
Session-Timeout                     = '0'
Unknown-Attribute                   = 'unknown code: attr {0,
197, 0, 4, "\0011-\000"}'
Acct-Output-Gigawords               = '0'
Acct-Authentic                      = 'RADIUS'
NAS-Port-Id                         = 'vlan140'
```

```
Framed-Protocol                   = 'PPP'
Event-Timestamp                   = '1514301672'
Acct-Input-Gigawords              = '0'
Ascend-Xmit-Rate                  = '20000000'
NAS-Port                          = '15760440'
```

Stop Accounting request

```
NAS-IP-Address                    = '41.213.1.32'
NAS-Port-Type                     = 'Ethernet'
User-Name                         = 'GP4899@ibhighspeed.co.za'
Called-Station-Id                 = 'pppoe-service1'
Calling-Station-Id                = 'E4:8D:8C:EE:1A:26'
Acct-Status-Type                  = 'Stop'
Service-Type                      = 'Framed-User'
Acct-Session-Id                   = '81707b87'
Acct-Input-Octets                 = '2668094'
Acct-Output-Octets                = '65615636'
Acct-Input-Packets                = '20697'
Acct-Output-Packets               = '52139'
Acct-Session-Time                 = '3725'
Acct-Delay-Time                   = '0'
Mikrotik-Realm                    = 'ibhighspeed.co.za'
Unknown-Attribute                 = 'unknown vendor: attr
{311, 10, 0, 17, "ibhighspeed.co.za"}'
Idle-Timeout                      = '86400'
Acct-Terminate-Cause              = 'Lost-Carrier'
NAS-Identifier                    = 'WDSL_PPPoE2_MDC'
Framed-IP-Address                 = '41.213.82.27'
Session-Timeout                   = '0'
Unknown-Attribute                 = 'unknown code: attr {0,
197, 0, 4, "\0011-\000"}'
Acct-Output-Gigawords             = '0'
Acct-Authentic                    = 'RADIUS'
NAS-Port-Id                       = 'vlan140'
Framed-Protocol                   = 'PPP'
Event-Timestamp                   = '1514303597'
Acct-Input-Gigawords              = '0'
Ascend-Xmit-Rate                  = '20000000'
NAS-Port                          = '15760440'
```

xDR import post-processing

Accounting request, originate leg

```
NAS-IP-Address = 127.0.0.101
NAS-Port-Id = IMPORT 01
User-Name = 01004
Called-Station-Id = 01020
Calling-Station-Id = 01004
Acct-Status-Type = Stop
PortaOne-Service-Type = Voice
h323-call-origin = originate
h323-call-type = Telephony
PortaOne-Service-Features = charge-report=1
h323-setup-time = 10:44:24.000 CEST Mon Mar 30 2009
h323-connect-time = 10:44:34.000 CEST Mon Mar 30 2009
h323-disconnect-time = 10:48:24.000 CEST Mon Mar 30 2009
h323-conf-id = 01010101 00000000 00000000 16000023
Acct-Session-Time = 230
Acct-Delay-Time = 86400
h323-ivr-out = PortaBilling_RatePattern:01020
```

```
h323-ivr-out = PortaBilling_Session:unlock
```

Accounting request, charge report:

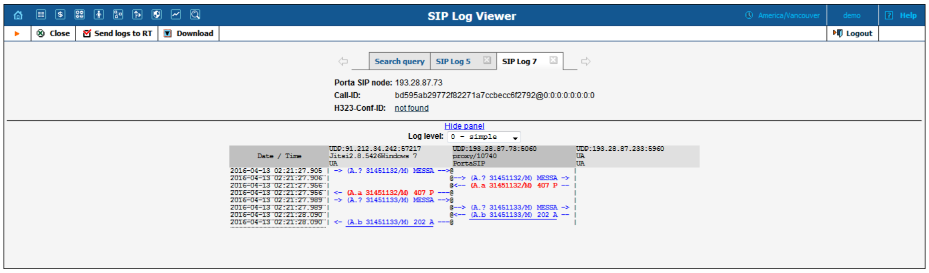
```
h323-ivr-in = PortaBilling_Session:id=01010101 00000000
00000000 16000023;i_service_type=3;i_env=3;valid=1
PortaOne-Charge = type=account;storage-type=Account;charged-
amount=0.00005;charged-quantity=10;i_account=500055
PortaOne-Charge = type=reseller;storage-type=Customer;charged-
amount=0.00010;charged-quantity=10;i_customer=20
PortaOne-Charge = type=vendor;storage-type=Vendor;charged-
amount=0.00005;charged-quantity=11;i_vendor=57
```

Examples of SIP communication

Messaging via SIP SIMPLE

In this example a message is sent between accounts of the same IP Centrex environment using classic (non cluster) PortaSwitch® configuration.

Part 1. Sending message



Porta SIP node: 193.28.87.73
Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
H323-Conf-ID: not found

Log level: 0 - simple

```
   Date / Time            UDP:91.212.34.242:57217   UDP:193.28.87.73:5060     UDP:193.28.87.233:5960
                          Jitsi2.8.5426Windows 7    proxy/10740               UA
                          UA                        PortaSIP                  UA
2016-04-13 02:21:27.905 | -> (A.? 31451132/M) MESSA -->@                      |
2016-04-13 02:21:27.906 |                            @--> (A.? 31451132/M) MESSA -> |
2016-04-13 02:21:27.956 |                            @<-- (A.a 31451132/M) 407 P -- |
2016-04-13 02:21:27.956 | <- (A.a 31451132/M) 407 P ---@                      |
2016-04-13 02:21:27.989 | -> (A.? 31451133/M) MESSA -->@                      |
2016-04-13 02:21:27.989 |                            @--> (A.? 31451133/M) MESSA -> |
2016-04-13 02:21:28.090 |                            @<-- (A.b 31451133/M) 202 A -- |
2016-04-13 02:21:28.090 | <- (A.b 31451133/M) 202 A ---@                      |
```

A sender's user agent sends a message to the PortaSIP® node

```
2016-04-
13T09:21:27.905890Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|622| RECEIVED message from UDP:91.212.34.242:57217
at UDP:193.28.87.73:5060:
     MESSAGE sip:7008004@193.28.87.73 SIP/2.0
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
ce7da5393ec80f60be280c053a117f13;received=91.212.34.242
     Max-Forwards: 70
     Contact: "7008001"
<sip:7008001@192.168.31.60:22430;transport=udp;registering_acc=193_28_8
7_73>
     To: <sip:7008004@193.28.87.73>
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451132 MESSAGE
     Content-Type: text/plain
     User-Agent: Jitsi2.8.5426Windows 7
     Content-Length: 19

     Hello! How are you?
```

The PortaSIP® node transfers the message to the IMGate node

```
2016-04-
13T09:21:27.906677Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|808| SENDING message to UDP:193.28.87.233:5960 from
UDP:193.28.87.73:5060:
     MESSAGE sip:7008004@193.28.87.233:5960 SIP/2.0
     Via: SIP/2.0/UDP 193.28.87.73:5060;branch=z9hG4bK-524287-1---
5d97e87d8fb47b33;rport
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
ce7da5393ec80f60be280c053a117f13;received=91.212.34.242;rport=57217
     Max-Forwards: 69
     Contact:
"7008001"<sip:7008001@91.212.34.242:57217;transport=udp;registering_acc
=193_28_87_73>
     To: <sip:7008004@193.28.87.73>
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451132 MESSAGE
     Content-Type: text/plain
     User-Agent: Jitsi2.8.5426Windows 7
     PortaSIP-Notify: NAT=91.212.34.242
     PortaOne-Requested-URI: sip:7008004@193.28.87.73
     Content-Length: 19

     Hello! How are you?
```

IMGate responds to the PortaSIP® node that user's authentication is required

```
2016-04-
13T09:21:27.956696Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|662| RECEIVED message from UDP:193.28.87.233:5960 at
UDP:193.28.87.73:5060:
     SIP/2.0 407 Proxy Authentication Required
     Via: SIP/2.0/UDP 193.28.87.73:5060;branch=z9hG4bK-524287-1---
5d97e87d8fb47b33;rport=5060
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
ce7da5393ec80f60be280c053a117f13;received=91.212.34.242;rport=57217
     Proxy-Authenticate: Digest
nonce="1460539287:cd0ab87b3060a8856d1558cf6233707d",algorithm=MD5,realm
="sip-41.TechWritting"
     To: <sip:7008004@193.28.87.73>;tag=3f3aff1b
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451132 MESSAGE
     Content-Length: 0
```

The PortaSIP® node sends authentication request to the sender's UA

```
2016-04-
13T09:21:27.956965Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|572| SENDING message to UDP:91.212.34.242:57217 from
UDP:193.28.87.73:5060:
     SIP/2.0 407 Proxy Authentication Required
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
ce7da5393ec80f60be280c053a117f13;received=91.212.34.242;rport=57217
     Proxy-Authenticate: Digest
nonce="1460539287:cd0ab87b3060a8856d1558cf6233707d",algorithm=MD5,realm
="sip-41.TechWritting"
     To: <sip:7008004@193.28.87.73>;tag=3f3aff1b
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451132 MESSAGE
     Content-Length: 0
```

The sender’s UA sends proxy authorization information to the PortaSIP® node

```
2016-04-
13T09:21:27.989028Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|839| RECEIVED message from UDP:91.212.34.242:57217
at UDP:193.28.87.73:5060:
     MESSAGE sip:7008004@193.28.87.73 SIP/2.0
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
67b34549a6d8b28b42f4488789779e2f;received=91.212.34.242
     Max-Forwards: 70
     Contact: "7008001"
<sip:7008001@192.168.31.60:22430;transport=udp;registering_acc=193_28_8
7_73>
     To: <sip:7008004@193.28.87.73>
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451133 MESSAGE
     Content-Type: text/plain
     Proxy-Authorization: Digest username="7008001",realm="sip-
41.TechWritting",nonce="1460539287:cd0ab87b3060a8856d1558cf6233707d",ur
i="sip:7008004@193.28.87.73",response="a9166ff5c75ae4fa9f6813fe581de2c4
",algorithm=MD5
     User-Agent: Jitsi2.8.5426Windows 7
     Content-Length: 19

     Hello! How are you?
```

The PortaSIP® node transfers the proxy authorization information message to IMGate

```
2016-04-
13T09:21:27.989781Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|1025| SENDING message to UDP:193.28.87.233:5960 from
UDP:193.28.87.73:5060:
     MESSAGE sip:7008004@193.28.87.233:5960 SIP/2.0
     Via: SIP/2.0/UDP 193.28.87.73:5060;branch=z9hG4bK-524287-1---
7e01222648b0c370;rport
     Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
67b34549a6d8b28b42f4488789779e2f;received=91.212.34.242;rport=57217
     Max-Forwards: 69
     Contact:
"7008001"<sip:7008001@91.212.34.242:57217;transport=udp;registering_acc
=193_28_87_73>
     To: <sip:7008004@193.28.87.73>
     From: "7008001" <sip:7008001@193.28.87.73>;tag=f2e03f39
     Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
     CSeq: 31451133 MESSAGE
     Content-Type: text/plain
     Proxy-Authorization: Digest username="7008001",realm="sip-
41.TechWritting",nonce="1460539287:cd0ab87b3060a8856d1558cf6233707d",ur
i="sip:7008004@193.28.87.73",response="a9166ff5c75ae4fa9f6813fe581de2c4
",algorithm=MD5
     User-Agent: Jitsi2.8.5426Windows 7
     PortaSIP-Notify: NAT=91.212.34.242
     PortaOne-Requested-URI: sip:7008004@193.28.87.73
     Content-Length: 19

     Hello! How are you?
```

IMGate sends the authorization request to the PortaBilling® and if the PortaBilling® authorizes the sender, informs the PortaSIP® node about this and that and the message is accepted

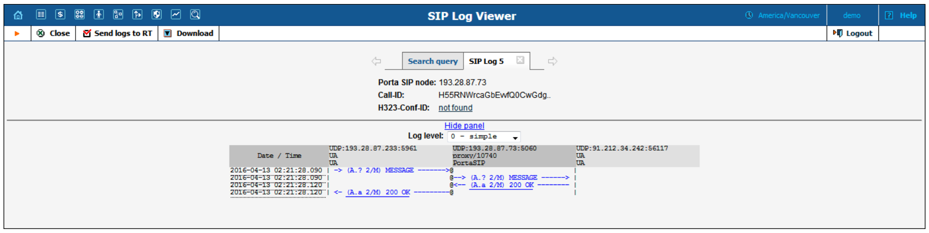
```
2016-04-
13T09:21:28.090083Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
```

```
0:0:0:0:0:0:0:0|IS|518| RECEIVED message from UDP:193.28.87.233:5960 at
UDP:193.28.87.73:5060:
    SIP/2.0 202 Accepted
    Via: SIP/2.0/UDP 193.28.87.73:5060;branch=z9hG4bK-524287-1---
7e01222648b0c370;rport=5060
    Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
67b34549a6d8b28b42f4488789779e2f;received=91.212.34.242;rport=57217
    To: <sip:7008004@193.28.87.73>;tag=95ab980f
    From: "7008001"<sip:7008001@193.28.87.73>;tag=f2e03f39
    Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
    CSeq: 31451133 MESSAGE
    Content-Length: 0
```

The PortaSIP® node informs the sender's UA that that the authorization is successful the message is accepted

```
2016-04-
13T09:21:28.090353Z|portaproxy[10740]|bd595ab29772f82271a7ccbecc6f2792@
0:0:0:0:0:0:0:0|IS|428| SENDING message to UDP:91.212.34.242:57217 from
UDP:193.28.87.73:5060:
    SIP/2.0 202 Accepted
    Via: SIP/2.0/UDP 192.168.31.60:22430;branch=z9hG4bK-333636-
67b34549a6d8b28b42f4488789779e2f;received=91.212.34.242;rport=57217
    To: <sip:7008004@193.28.87.73>;tag=95ab980f
    From: "7008001"<sip:7008001@193.28.87.73>;tag=f2e03f39
    Call-ID: bd595ab29772f82271a7ccbecc6f2792@0:0:0:0:0:0:0:0
    CSeq: 31451133 MESSAGE
    Content-Length: 0
```

Part 2. Delivering message



Porta SIP node: 193.28.87.73
Call-ID: H55RNWrcaGbEwfQ0CwGdg..
H323-Conf-ID: not found

Log level: 0 - simple

```
               UDP:193.28.87.233:5961        UDP:193.28.87.73:5060          UDP:91.212.34.242:56117
 Date / Time   UA                            proxy/10740                    UA
               UA                            PortaSIP                       UA
2016-04-13 02:21:28.090 | -> (A.? 2/M) MESSAGE -------->@                   |
2016-04-13 02:21:28.090 |                              @--> (A.? 2/M) MESSAGE ------> |
2016-04-13 02:21:28.120 |                              @<-- (A.a 2/M) 200 OK -------- |
2016-04-13 02:21:28.120 | <- (A.a 2/M) 200 OK ---------@                    |
```

IMGate transfers the message to a recipient's SIP server

```
2016-04-
13T09:21:28.090215Z|portaproxy[10740]|H55RNWrcaGbEvvfQ0CwGdg..|IS|500|
RECEIVED message from UDP:193.28.87.233:5961 at UDP:193.28.87.73:5060:
    MESSAGE sip:7008004@193.28.87.73:5060 SIP/2.0
    Via: SIP/2.0/UDP 193.28.87.233:5961;branch=z9hG4bK-524287-1---
834c3c3758bb016b;rport=5961
    Max-Forwards: 70
    Route: <sip:193.28.87.73:5060;lr>
    To: <sip:7008004@193.28.87.73:5060;lr>
    From: <sip:7008001@193.28.87.233:5961>;tag=5b0b2f73
    Call-ID: H55RNWrcaGbEvvfQ0CwGdg..
    CSeq: 2 MESSAGE
    Allow: MESSAGE, NOTIFY
    Content-Type: text/plain
    Content-Length: 19

    Hello! How are you?
```

The recipient's SIP server transfers the message to the recipient's UA

```
2016-04-
13T09:21:28.090539Z|portaproxy[10740]|H55RNWrcaGbEvvfQ0CwGdg..|IS|569|
SENDING message to UDP:91.212.34.242:56117 from UDP:193.28.87.73:5060:
    MESSAGE sip:7008004@91.212.34.242:56117;ob;transport=UDP SIP/2.0
    Via: SIP/2.0/UDP 193.28.87.73:5060;branch=z9hG4bK-524287-1---
45fb5d774b48af55;rport
    Via: SIP/2.0/UDP 193.28.87.233:5961;branch=z9hG4bK-524287-1---
834c3c3758bb016b;rport=5961
    Max-Forwards: 69
    To: <sip:7008004@193.28.87.73:5060;lr>
    From: <sip:7008001@193.28.87.233:5961>;tag=5b0b2f73
    Call-ID: H55RNWrcaGbEvvfQ0CwGdg..
    CSeq: 2 MESSAGE
    Allow: MESSAGE, NOTIFY
    Content-Type: text/plain
    Content-Length: 19

    Hello! How are you?
```

The recipient's UA responds to the recipient's SIP server that the message is accepted

```
2016-04-
13T09:21:28.120507Z|portaproxy[10740]|H55RNWrcaGbEvvfQ0CwGdg..|IS|486|
RECEIVED message from UDP:91.212.34.242:56117 at UDP:193.28.87.73:5060:
    SIP/2.0 200 OK
    Via: SIP/2.0/UDP
193.28.87.73:5060;rport=5060;received=193.28.87.73;branch=z9hG4bK-
524287-1---45fb5d774b48af55
    Via: SIP/2.0/UDP 193.28.87.233:5961;rport=5961;branch=z9hG4bK-
524287-1---834c3c3758bb016b
    To: <sip:7008004@193.28.87.73>;tag=z9hG4bK-524287-1---
45fb5d774b48af55
    From: <sip:7008001@193.28.87.233>;tag=5b0b2f73
    Call-ID: H55RNWrcaGbEvvfQ0CwGdg..
    CSeq: 2 MESSAGE
    Content-Length: 0
```

The recipient's SIP server informs IMGate that the mesage is accepted

```
2016-04-
13T09:21:28.120760Z|portaproxy[10740]|H55RNWrcaGbEvvfQ0CwGdg..|IS|372|
SENDING message to UDP:193.28.87.233:5961 from UDP:193.28.87.73:5060:
    SIP/2.0 200 OK
    Via: SIP/2.0/UDP 193.28.87.233:5961;rport=5961;branch=z9hG4bK-
524287-1---834c3c3758bb016b
    To: <sip:7008004@193.28.87.73>;tag=z9hG4bK-524287-1---
45fb5d774b48af55
    From: <sip:7008001@193.28.87.233>;tag=5b0b2f73
    Call-ID: H55RNWrcaGbEvvfQ0CwGdg..
    CSeq: 2 MESSAGE
    Content-Length: 0
```

Description of xDR record fields

Field	Type	Description
id	int	The unique ID of the xDR
i_env	int	The unique ID of the virtual billing environment
h323_conf_id (session_id)	string	The unique session ID
call_id (service_session_id)	string	The session ID specific for the xDR's service type

CLI (cli)	string	The caller number (also called CLI or ANI)
CLD (cld)	string	The destination number (also called CLD, Called-Station-Id or DNIS)
setup_time	int	Also known as **PDD**. The time (in ms) between the moment the first INVITE is sent in the originate leg and the moment a provisional response/final response comes from the SIP device. It is calculated by the Billing Engine as the time difference between *h323-setup-time* and *alert-timepoint* (h323-connect-time is used instead of alert-timepoint, if the latter is not present in the accounting request)
connect_time	dateTime	The time when the session started (e.g., the call got connected)
disconnect_time	dateTime	The time when the session was disconnected or was last updated (for incremental xDR charge)
bill_time	dateTime	Call bill time
disconnect_cause	int	Specifies the reason the call was disconnected. This is a hexadecimal code
voice_quality	int	Shows call quality provided during the session as/if reported by NAS (provided for voice calls by Cisco equipment)
used_quantity	int	The quantity that was charged in original measurement units (e.g. duration of the voice call in seconds)
account_id	string	ID of the account being charged
i_account	int	The unique ID of the account's record in the database
i_dest	int	The unique ID of the destination record
i_dest_group	int	The unique ID of the destination group
i_tariff	int	The unique ID of the tariff record
i_rate	int	The unique ID of the rate record
charged_quantity	int	The number of charged units (the service is measured based on *used_quantity* after applying rounding intervals)

charged_amount	float	The amount charged in the respective currency
history	string	The formula that shows charge application steps or a textual description of the charge applied
peak_level	int	The ID of the peak level: 0 (peak period), 1 (off-peak period), 2 (2nd off-peak period)
i_vendor	int	Applicable for vendor xDRs. The unique ID of the Vendor record who was billed
i_service	int	The unique ID of the service record
i_accessibility	int	The unique ID of the rating entry record in the database
i_customer	int	The unique ID of the customer or the reseller (where applicable) associated with the call. This ID should be equal to the corresponding account's owner in CDR_Accounts, to the xDR owner in CDR_Customers and to the direct customer in CDR_Vendors
h323_incoming_conf_id (parent_session_id)	string	The unique ID of the incoming session (if exists) used for interrelating xDRs, when the charged session is established as a result of a previous session (possibly having its own xDR)
rating_pattern	string	The rate pattern that was used to match a destination in the tariff rates
*subscriber*_ip	string	The internet address of the end-user (if applicable) which was assigned to him in the moment of charge application
split_order	int	The xDR order number in case it is a part of the group of xDRs created for the same session
billing_model	int	Specifies the account type; one of the following: -1 – Debit account 1 – Credit account
i_invoice	int	The unique ID of the invoice record. Designates the invoice where the xDR was settled
revenue	float	Applicable for vendor xDRs. Shows total revenue collected for the related session