CENTRAL CONTROL UNIT CLIENT & SERVER

PosConneXion CCU API

Release	Date	Author	Modification
1.10	2021-08-24	Peter Andersson, Infrasec Sweden	New version

1 Confidentiality clause

This document is an internal document of Infrasec Sweden AB and may not be redistributed, copied or changed in any form without a written permission of Infrasec Sweden AB.

2 Purpose

This document describes how to integrate with Infrasec Central Control Unit for registers from a server based retail system.

3 Limitations

This document tries to describe the main functionality of Infrasec Central Control Unit API but does not necessarily describe all features and details of the latest release. Information in this document may change and validity may be restricted without further notice.

4 iSecure component family

The Infrasec Central Control Unit is developed using technology of the Infrasec iSecure Security Component family. The Infrasec iSecure components are released on Solaris, Linux and WIN32 platforms. Other platforms are supported on demand. All components are configured using XML and DTD validation preventing syntax errors. The components share a foundation configuration that ensures conformity between Infrasec iSecure components.

The foundation configuration supply distributed logging that allows for log records to be concentrated at remote nodes and allows for logging at multiple devices and in different languages. The well design C-code ensures maximum performance and the reusable component architecture ensures secure and reliable code. The components share a built in TRACE and DEBUG functionality that ensures fast error detection and error corrections in delivered software components executing at customer site. For encryption the components support HSM support: Key Storage, RSA acceleration, "True" Random for SSL negotiation and encrypted ticket

5 Terminology

Term	Definition

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

6 Overview

This document describes the POS conneXion CCU API and is aimed for POS Application Vendor developers that enable provided CCU services in their POS Application. The POS conneXion API supply Retailers/Merchants an easy access to enroll a specific POS in the Infrasec POS conneXion Service Cloud.

Current POS conneXion API services:

This release

- Central Tax Control Unit
- Swish Corporate

Pipeline

- Distributed Order with Clearing Services
- Distributed Voucher Balance with Clearing Services
- Distributed Electronic Receipt Storage

The Infrasec Central Control Unit (CCU) API enables individual registers and register servers on behalf of its manged registers to execute using SKVFS Swedish Tax Authority control codes from Infrasec central control unit. Infrasec CCU API enable distributed registers secure connections to the central located CCU.

Infrasec CCU API is supplied together with X.509 certificates to CCU customers that are used to identify and secure transactions for a single register, a specific register server, a organization unit or a full organization.

Infrasec configure and distribute PHP/C# examples to POS vendors that can be used to simplify the development for the vendor. The examples are ready to run where the vendor can send a investigate the communication flow between the POS services and the CCU.

Infrasec CCU API validates receipt content to conform to SKVFS regulations and enables secure access to the CCU using the supplied X.509 certificate.

The register application that is using the CCU API package the receipt content in a simple XML object as described in this document and issue a Infrasec CCU API function call or a HTTP post to the Infrasec IPC Service with the receipt XML object as body. The receipt XML object is validated to ensure SKVFS format and securely forwarded to the CCU using TLS authenticated by the X.509 certificate.

The CCU handles several organization entities and organization numbers owning registers. Each register must be named uniquely within the CCU. Infrasec issue register identities to CCU customers conforming to SKVFS format that are guaranteed to be unique.

In the most unusual event of communication problems where both primary and secondary options such as fixed lines xDSL, Fiber, Cable etc and or mobile such as 3G, 4G, 5G communication paths are unavailable, the register enter offline mode with unique receipt number series doing the following:

- Issue cash invoices, kontantnota (Already accepted by the tax authority)

The communication to the CCU is part of Infrasec PosConneXion framework where a number of optional and additional services are available. This specification defines the XML format that must be followed when issuing a CCU request to bundled in the PosConnexion request. The PosConneXion bundling and how to tag the request as a CCU request is described in the PosConneXion API specification.

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

7 Infrasec CCU API Methods

Infrasec supply different methods to integrate with the CCU where communication, register environment, sales application and customer organization impose what is preferred. In the context below we mean Client when the register as a client application calls the CCU API. With server we mean that the register application is centralized and that a application server calls the CCU API on behalf of the register.

The Infrasec PosConneXion CCU API methods:

- TLS/HTTPS server requests sent from partner server that acts as a agent to the CCU for the POS Device
- TLS/HTTPS server requests sent from merchant server acting as a agent to the CCU for the POS Device
- TLS/HTTPS client requests sent directly from POS Device to the CCU
- iSecureFMK API Windows DLL or Unix Shared Object (Client or server) Proxy
- iSecureIPC API is a Socket Gateway Windows service or Unix Proxy application (Client or server)

The request to the CCU is always sent over TLS/HTTPS but the preferred method may differ depending on the particular requirement and implementation of the POS application. The POS application may include the required certificates locally in the application, in a local or remote proxy server or by including Infrasec DLL/SO in the POS application or server. The optimal communication solution for your POS services including certificate management is selected in cooperation with Infrasec.

The iSecureFMK API is loaded into the application by either LoadLibrary() or dlopen() depending on operating system. The appplication initiates the iSecureFMK API by calling iAppInit() during application startup and calls iAppExit() before application termination. To request a receipt control code from the CCU the application calls the function iGetReceiptCode().

The iSecureIPC API consists of a gateway process that listens on a customer application server for a localhost port for receipt requests and forwards the receipts to the CCU. To integrate this method the register application execute a TCP connect to the localhost based port and sends a HTTP POST on the socket with the recipt XML object as a body. The response to the POST is a HTTP response with the body consisting of the response XML object.

8 Client or Server CCU API

Depending on the selected integration method the CCU may be accessed either from the client register directly or from a server that acts on behalf of the registers. For the CCU API itself very little differ. This is mainly a decision for the sales application.

For client based CCU integration every register accessing the CCU requires a unique X.509 certificate (enrollment) for the TLS communication. This means that the mechanism to support X.509 certificates enrollment not can (or is recommended) to be handled manually. For the best methods to handle certificate enrollment in this particular case please contact Infrasec Sweden AB

For server based CCU integration only one customer X.509 certificate is required for the TLS communication / protected unit such as a server or organization etc. This means that the X.509 certificates normally can be handled manually not requiring automated enrollment.

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

9 Receipt XML format

Before completing a sale and writing a receipt the register application builds a CCU Receipt Request XML object that is used when calling the CCU API. The CCU API validates the content and using TLS and the X.509 certificate sends the XML to the CCU. The CCU validates that the used X.509 certificate holds the rights to operate the register and creates a receipt response with control code that is returned back through the CCU API. The register application stores the control code together with the receipt in the receipt database and completes the sale and writes the receipt

A receipt request XML contains the following:

TAG	Content	Comment
<ApplicationID>	64 Alphanumeric characters	Value is returned in response. Register application use this value to identify requests from application instances or versions
<RequestID>	64 Alphanumeric characters	Value is returned in response. Register application use this value to identify individual requests
<DateTime>	12 Numeric characters: Format: YYYYMMDDttmm	Date and time of the minute the receipt request is generated. Ex 201012241945
<OrgNr>	10 Numeric characters	Organization number of the selling organization using the register in this request
<ManRegisterID>	16 Alphanumeric characters	Restricted: Unique register identity of this register and only used by one organization. Ex: EP01AB0101330001
<SequenceNumber>	12 Numeric characters	Receipt sequence number
<ReceiptType>	- normal - kopia - ovning - profo	Matching the receipt type for which this receipt request is executed. Ex: <ReceiptType Type="normal"/>
OPTION: <SaleAmount>	11 ',' 2 Numeric characters	A receipt must only contain Sale or Refund. Ex: <SaleAmount>100,00</SaleAmount>
OPTION: <RefundAmount>	11 ',' 2 Numeric characters	A receipt must only contain Sale or Refund. Value is the absolute value of the refund amount Ex: <RefundAmount>200,00</RefundAmount>
<VAT1> .. <VAT4>	<Percent> and <Amount>	All four VAT sections must be available even though they not contain any percent or amount other then 0,00
<Percent>	2','2 Numeric characters	Normally 25, 12, 6 or 0 percent Ex: <Percent>25,00</Percent> <Percent>12,00</Percent> <Percent>6,00</Percent> <Percent>0,00</Percent>
<Amount>	11','2 Numeric characters '-'10','2 Numeric characters	VAT amount on sale is positive. VAT amount on refund is negative

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

A receipt response XML contains the following:

TAG	Content	Comment
<ResponseCode>	Response result code	Numeric value indicating response error code or success code. Success code is 0.
<ResponseMessage>	Response result text	Response code translated into message text
<ResponseReason>	Response result reason	Further information text describing the reason for a non success error response code
<ApplicationID>	64 Alphanumeric characters	Restricted: Value returned from request. Register application use this value to identify requests from application instances or versions
<RequestID>	64 Alphanumeric characters	Restricted: Value returned from request. Register application use this value to identify individual requests
<CTUID>	17 Alphanumeric characters	The identity of the CCU (Central Tax/Control Unit) to be written on the receipt together with the register identity
<Code>	59 Alphanumeric characters	The control code to be stored together with the receipt data (database field) by the sales application.

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

10 The POS conneXion API

The POS conneXion API is built to handle provided services through a generic list of extensions. A Request must always contain a header followed by a list of details for requested services. The Response always contains a header with general error information and a list of services holding details for concerned services.

10.1 XML POS conneXion Request Example

The Request contains a header section with general information and a list of extensions:

```
<posConneXionRequest>
  <ApplicationID>ACME POS v1.0</ApplicationID>
  <RequestID>806e589fc9506474</RequestID>
  <OrgNr>1212121212</OrgNr>
  <RegisterId>EX11212121212101</RegisterId>
  <DateTime>20170131142620</DateTime>
  <posConnexionServiceList>
    <posConnexionService>
      <ServiceID>CCU</ServiceID>
      <ServiceVersion>1.0</ServiceVersion>
      <ServiceFormat>application/xml</ServiceFormat>
      <ServiceEncoding>base64</ServiceEncoding>
      <ServiceData>BASE64XXX</ServiceData>
    </posConnexionService>
  </posConnexionServiceList>
</posConneXionRequest>
```

10.2 XML POS conneXion Response Example

The response contains a header section including eventual error information and a list of extensions:

```
[
<posConneXionResponse>
  <TransactionID>18198178230534683852</TransactionID>
  <ApplicationID>ACME POS v1.0</ApplicationID>
  <RequestID>806e589fc9506474</RequestID>
  <OrgNr>1212121212</OrgNr>
  <RegisterId>EX11212121212101</RegisterId>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <ResponseReason></ResponseReason>
  <posConnexionServiceList>
    <posConnexionService>
      <ServiceID>CCU</ServiceID>
      <ResponseCode>0</ResponseCode>
      <ResponseMessage>Success</ResponseMessage>
      <ResponseReason></ResponseReason>
      <ServiceData>BASE64XXX</ServiceData>
    </posConnexionService>
  </posConnexionServiceList>
</posConneXionResponse>
```


Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

10.3 General Heading Attributes

10.3.1 ApplicationID

Application ID is an information that represents the calling application. This is normally the application identification and version info of the POS / Register issuing the request. Information is returned in response.

10.3.2 RequestID

Request Id is information that the marks this request individually. This is normally a sequence number or a guid that uniquely identifies this request. Information is returned in response.

10.3.3 OrgNr

Organization number of the merchant / store responsible for the register issuing the request.

10.3.4 RegisterID

Registered register identity of the POS / Register issuing the request

10.3.5 DateTime

The date and time retreived by the POS / Register when the request is issued.

10.4 General Service Attributes

The POS conneXion API provides a general set of attributes that are common for all services. Information that is specific to the particular service must be passed as a base64 encoded object.

10.4.1 ServiceID

This field identifies the requested service that shall process this entry. The information is authorized that the POS / Register is allowed to communicate with the requested service, determines endpoint

10.4.2 ServiceVersion

This field defines the version the client executes, determines endpoint

10.4.3 ServiceFormat

This field is application/xml (Certain services supports application/json)

10.4.4 ServiceEncoding

This field is base64

10.4.5 ServiceData

This field contains a base64 encoding of the data to be processed by the requested service.

10.4.6 Internal: ServiceForward

This section is for POS conneXion Service Providers as described in General Service Forward Attributes

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

10.5 General Service Response Attributes

The POS conneXion API provides a general set of attributes that are common for all services. Information that is specific to the particular service must be passed as a base64 encoded object.

10.5.1 ServiceID

This field identifies the requested service that has processed this entry.

10.5.2 ResponseCode

This field identifies a numerical value for the success of this request. Value 0 means success

10.5.3 ResponseMessage

This field will contain a textual explanation corresponding to the response code above, Value Success is ok

10.5.4 ResponseReason

This may contain a textual explanation of the reason a failing request.

10.5.5 ResponseAlert

10.5.6 ServiceData

This field contains a base64 encoding of the data to be processed by the requested service.

10.5.7 Internal: ServiceForward

This section is for POS conneXion Service Providers as described in General Service Forward Attributes

10.6 General Service Forward Attributes

This section is for POS conneXion Service Providers as described in General Service Forward Attributes

10.6.1 Internal: ServiceForwardID

This identifies the target ServiceForwardID this information is allowed to be forwarded to.
Information in this field is made available to POS conneXion Service Providers.

10.6.2 Internal: ServiceForwardData

This field contains base64 of data forwarded to target service identified by the ServiceForwardID above. Information in this field is made available to POS conneXion Service Providers.

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

11 Receipt and control code examples

11.1 Receipt request example

Below is a receipt request for a receipt sent by register identity (Kassabeteckning) RA01AB0101330001 of the organization number 1111111111. The receipt is for a sale/purchase operation PurchaseAmount (normal) of 100,00 kronor. The receipt sequence number (Löpnummer) is 1001 and VAT list is a single VAT of 25,00 percent and 25,00 kronor.

The ApplicationID and RequestID values are optional to the API but is recommended though they are helpful when tracking individual requests

The ControlData corresponds to the receipt data (kvittodata) that are used when creating the SKVFS control code.

```
<CtuRequest>
  <ApplicationID>RegApp123</ApplicationID>
  <RequestID>RegApp123_1001</RequestID>
  <ControlData>
    <DateTime>201012241945</DateTime>
    <OrgNr>1111111111</OrgNr>
    <ManRegisterID>RA01AB0101330001</ManRegisterID>
    <SequenceNumber>1001</SequenceNumber>
    <ReceiptType Type="normal"/>
    <SaleAmount>100,00</SaleAmount>
    <VAT1>
     <Percent>25,00</Percent>
     <Amount>20,00</Amount>
    </VAT1>
    <VAT2>
     <Percent>12,00</Percent>
     <Amount>0,00</Amount>
    </VAT2>
    <VAT3>
     <Percent>6,00</Percent>
     <Amount>0,00</Amount>
    </VAT3>
    <VAT4>
     <Percent>0,00</Percent>
     <Amount>0,00</Amount>
    </VAT4>
  </ControlData>
</CtuRequest>
```

Refund Note:
On a refund the <PurchaseAmount> field is replaced with a <RefundAmount> field with the absolute value (positive) of the refund amount. The VAT fields holding a <Amount> field different from 0,00 must be negative as in the example here <Amount>-20,00</Amount>

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

11.2 Receipt response example

Below is a receipt response for receipt sent using request identity `RegApp123_1001`. The response is a success holding response code 0.

The response contains the CTUID `ISCCUSE0000000000` of 17 characters that must be written on the receipt and the control code `Code` of 59 characters that shall be stored together with the receipt in the receipt database.

The ApplicationID and RequestID values are returned as from the request

```
<CtuResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <ResponseReason></ResponseReason>
  <ApplicationID>RegApp123</ApplicationID>
  <RequestID>RegApp123_1001</RequestID>
  <ControlCode>
    <CTUID>ISCCUSE0000000000</CTUID>
    <Code>JC3VIWNLDPZNM3FE6C5AUW4ITVAALKXD;6RZAYXAANQU5C5FUWNREDO2EP4</Code>
  </ControlCode>
</CtuResponse>
```

11.3 Receipt response error

Below is a receipt response for receipt sent using request identity `RegApp123_1001`. The response is a error where the register organization identity is incorrect.

Error responses hold a `ResponseCode` different from 0 and a `ResponseMessage` describing the error and a further explanation in the `ResponseReason` field.

The ApplicationID and RequestID values are returned as from the request

On errors sale MUST not be completed. Receipt for a sale MUST not be written.

```
<CtuResponse>
  <ResponseCode>130</ResponseCode>
  <ResponseMessage>
   The receipt CTU server is not accepting the receipt
  </ResponseMessage>
  <ResponseReason>
   The referenced organization identity is incorrect
  </ResponseReason>
  <ApplicationID>RegApp123</ApplicationID>
  <RequestID>RegApp123_1001</RequestID>
  <ControlCode>
    <CTUID>ISCCUSE0000000000</CTUID>
  </ControlCode>
</CtuResponse>
```


Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

12 iSecureIPC API

The iSecureIPC API is available through a local application (service or daemon) that is installed in the register application server or close to it. Normally the iSecureIPC application is installed on the same server as the register application. The iSecureIPC application accepts a new TCP connect for every new receipt sent by the register application. The receipt is then validated to conform to SKVFS format and if the validation fails to validate a locally generated response is returned. If the validation succeeds a TLS session is established and the receipt is forwarded to be processed by the CCU. If the used X.509 certificate is authorized to communicate receipts for the used register the receipt content is processed by the CCU and a control code is returned. If any of the steps is not authorized a error XML is returned.

Register application must be able to setup a new socket for each receipt and issue a HTTP post but are not mandated to handle TLS. Register receipt requests communicated through iSecureIPC are forwarded securely using TLS identified by the X.509 certificate. The register application only act on clear text data. Encryption and security is handled by the CCU API.

The below HTTP request is for a server based request to be authorized through the URL /ctuserver at CCU server side. The content is a XML body and the length (before we included some new lines) is 594 bytes. Also note that the requested HTTP session shall be closed after completion.

This is a HTTP example request to iSecureIPC

```
POST /ctuserver HTTP/1.1
User-Agent: custombroker
Host: 127.0.0.1
Content-type: text/xml; charset=utf-8
Content-length: 594
Connection: Close
Cache-Control: no-cache

<CtuRequest>
<ApplicationID>REG APP 1.0.0.0</ApplicationID>
<RequestID>RA_VERIFY_RA01AB0101330001_1234</RequestID>
<ControlData>
<DateTime>201012241200</DateTime>
<OrgNr>1212121212</OrgNr>
<ManRegisterID>RA01AB0101330001</ManRegisterID>
<SequenceNumber>13416</SequenceNumber>
<ReceiptType Type="normal"/>
<PurchaseAmount>262,00</PurchaseAmount>
<VAT1><Percent>25,00</Percent><Amount>52,40</Amount></VAT1>
<VAT2><Percent>12,00</Percent><Amount>0,00</Amount></VAT2>
<VAT3><Percent>6,00</Percent><Amount>0,00</Amount></VAT3>
<VAT4><Percent>0,00</Percent><Amount>0,00</Amount></VAT4>
</ControlData>
</CtuRequest>
```


Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

The below HTTP response was a success containing 200 OK response status and the ResponseCode of 0. The content is a XML body and the length (before we included some new lines) is 363 bytes. Also note that the responded HTTP session shall be closed after completion.

This is a HTTP example response through iSecureIPC

```
HTTP/1.1 200 OK
Server: InfraSec iSecureServer
Content-Length: 363
Connection : Close
Content-Type: text/xml; charset=utf-8
Cache-Control: no-cache

<CtuResponse>
<ResponseCode>0</ResponseCode>
<ResponseMessage>Success</ResponseMessage>
<ResponseReason></ResponseReason>
<ApplicationID>REG APP 1.0.0.0</ApplicationID>
<RequestID>RA_VERIFY_RA01AB0101330001_1234</RequestID>
<ControlCode>
<CTUID>ISCCUSE0000000000</CTUID>
<Code>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX;YYYYYYYYYYYYYYYYYYYYYYYYY</Code>
</ControlCode>
</CtuResponse>
```


Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

12.1 HTTP Response Codes In CCU/IPC Communication

When a CCU request is sent to the CCU Server directly or via a locally managed IPC proxy a CCU error response may be communicated using HTTP Response Headers or within the responded XML body.

A CCU error may be communicated within a HTTP 200 OK reponse with an error reported in the ResponseCode element within the XML body or at the HTTP protocol layer with a HTTP Header response code apart from 200 OK.

This also means that the only situation where a CCU response code is successful is where the HTTP response is returned with HTTP 200 OK and the embedded XML body ResponseCode is 0 for Success.

Below is a explanation of HTTP header response codes

12.1.1 HTTP/1.1 200 OK

This is a processed HTTP response. This means that the CCU request has been processed such that the HTTP response will contain a XML body with a ResponseCode that informs the caller of the success or failure of the executed CCU request.

The XML body ResponseCode must be investigated for its status before continuing and creating the receipt.

12.1.2 HTTP/1.1 400 Bad Request

This is a general authentication error concerning used credentials to gain access to the CCU.

12.1.3 HTTP/1.1 403 Forbidden

This is an authorization or protocol error to gain access to the CCU for this register. This means that the request to gain access is malformed or that the requested TLS session lacks correct authorization to the requested service, in this case the service is identified by the requested url.

12.1.4 HTTP/1.1 412 Precondition Failed

This is a semi general authentication error where credentials are not registered, missing or not registered to gain access to the CCU. Credentials in this text mean that either the organization number or the register identity either is missing or not activated. The token represented by used client certificate for the accessed TLS session may also not be trusted, expired, closed or revoked.

12.1.5 HTTP/1.1 901 Secret

The register client secret is not validated successfully

12.1.6 HTTP/1.1 902 Revoked

The register client certificate is revoked for access to the CCU

12.1.7 HTTP/1.1 903 Blocked

The register client certificate is disabled for access to the CCU

12.1.8 HTTP/1.1 904 Disabled

The register client certificate is disabled for access to the CCU

12.1.9 HTTP/1.1 920 CTU Unavailable

The receipt could not reach CCU due to that the CCU is offline:

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se

12.1.10 HTTP/1.1 921 CTU Reject

The request is rejected from processing through the CCU due to that:

- Register is denied access to the CCU
- Receipt content is not accepted by the CCU

13 ResponseCode

These responsecodes are the only responses that are allowed when creating receipts. For the full list of possible response and error codes please refer to the iMessagId.h header file.

Code	Comment
0	Success

Infrasec Sweden AB
Visit address: Rosengatan 8, Sundbyberg
Invoice address: faktura@infrasec.se

Telephone
+46 (0) 8 411 12 40

Bankgiro
5396-5901

Org.nr.
556624-1674

URL
www.infrasec.se