IDM & CCU

PosConneXion

ENROLLMENT API

Administration Services

Release	Date	Author	Modification
Rel. 1.23	2021-11-08	Peter Andersson, Infrasec Sweden	New Version

1 Confidentiality clause

2 Purpose

This document describes how to integrate with Infrasec PosConneXion and Central Control Unit enrollment services API to automate the process to manage POS registers. Trusted partners may through this API from their back-office registration servers create and manage partner POS identities with related information for related services.

3 Limitations

This document tries to describe the main functionality of PosConneXion and Infrasec Central Control Unit Register Enrollment 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 PosConneXion and 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
CCU	Infrasec ISCCU Central Tax Control Unit
IDM	Infrasec Identity Management control access for partner POS to PosConneXion services
IDM Tree	IDM is organized in a tree structure. Administration can only operate subordinate objects
IDM Authority	IDM Tree base node for a Partner where partner prefix and distribution model are defined
IDM POS Authority	IDM POS Authority defines Prefix structure for how RegisterID's are constructed.
IDM Prefix	IDM Prefix is a code that in the IDM tree, when used is IDM unique for the particular node
IDM Path	The path (DN) to a node or object in the IDM Tree. The path is given by / separator of the intermediate node names (RDN) from the IDM Authority base node
PosConneXion (PCX)	Infrasec POS Connection API providing access to managed services (CCU/SWISH/..)
POS RegisterID	The Infrasec Global Unique Identity acquired to a POS register during enrollment
POS Register	Partner POS identified by its unique RegisterID (will also be registered with Swedish TAX)
POS Agent Certificate	Partner identity certificate authorized to send (PCX) requests for tree subordinate POS:es
POS Certificate	Partner identity certificate authorized to send (PCX) requests for itself (given RegisterID)
LRA Administrator	Local Registration Authority API identity having IDM Access to manage POS services

6 Overview

The main purpose of the Enrollment API is to enable functionality for a partner/vendor to automate the management of identities in a structured and auditable process to ensure correct register and services registration.

The Infrasec PosConneXion and CCU enrollment process allows for management of register identities for the access control to the CCU and related services. For CCU also as for manual enrollment the automated communication of the resulting management of these identities to the Swedish Tax Authority is included.

A Partner creates own register identities and is handled as an Authority that owns a unique namespace under which these identities can be created. An Authority is identified by an Authority ID that normally is two to four characters that also are the first characters in the managed identities.

Infrasec Enrollment API is supplied together with one or more unique X.509 client certificates and eventual Smartcards for HSM usage that are given a POS/Register partner authority privilege that allows for management of the partners register identities. Mostly Smartcards or Soft Certificates in combination with SMS Code are used for administrator login to the CCU Identity Management GUI. All actions through the Enrollment API and executed through the IDM GUI are audited.

The partner application that is using the Register Enrollment API executes HTTPS posts using TLS secured communication paths authorized for the X.509 certificate to the Infrasec Enrollment Process with the new or modified register information in XML or JSON object body. The register information is validated to ensure proper authorization with the Swedish Tax Authority and the Access Control Policy before the process is continued.

The PosConneXion and CCU Access Control Policy handles several partners, organization entities and organization numbers owning registers. Each register is named uniquely within the CCU (Tax Authority Regulation) why Infrasec issue register identities to CCU partners customers conforming to SKVFS format that are guaranteed to be unique within PosConneXion and CCU.

A partner is given a unique naming space under which the partner customers Register Identities are created. Each register identity consists of sixteen characters and a partner naming space consists of two to four leading characters. The following twelve characters of the register identity are constructed as agreed between Infrasec and the partner.

Register identities are created in a hierarchical tree structure such that a Partners tree structure contain organization branches that can be combined differently and may be that line of businesses contains chains that in turn contain stores that contain registers. How this structure is created and managed is decided in agreement between the partner and Infrasec. Each node in the tree may contribute with a specific set of characters in the process of creating the unique register identity.

The Enrollment API enroll and onboard concerned register to required PosConneXion services. The enrollment to the service is automatic and the service can be used as soon as the Enrollment API call is completed. Certain services requires that the concerned Store in advance has an agreement with the service provider, as a Swish Nr with a Bank

7 Requirements

7.1 Usage

System interface for enrollment of POS for CCU slot usage. The information must contain correct store information a stated in Swedish Tax Authority regulations. The POS vendor is responsible that the Store/POS information is correct. Only approved system is allowed to access the service.

7.2 Registration Information

Usage of CCU enrollment service is restricted to POS vendors that guarantees the correctness of enrollment information and have authorization from the end user to register correct company information. All company information that is given in an enrollment must be correct and verified against Swedish "Bolagsverket" or equivalent.

7.3 TAX Authority

Executing API calls for a register CCU Slot most probably means that information for this register is communicated to the TAX Authority.

The API Methods NEW and OPEN results in that the TAX authority will send out new information to the register organization that the register is active.

7.4 CCU Slot

An enrollment request, register and allocates a CCU slot that is unique for a specific register. The CCU slot can't be reused by another register. It's allowed to set a POS in CLOSE status if the register isn't in usage.
To prevent miss usage and frauds: During a unique register lifecycle.

7.5 Swish Service

The Store or Web Server/Shop Service that is enrolled to Swish services must have a completed agreement with the Bank and hold a Swish Nr before enrolling to the Swish service. When enrollment is completed Swish payments to the Swish Nr is available for the enrolled POS register/Web Server.

7.6 Digital Receipt Service

Enrollment to the Digital Receipt Service will enroll the store to the Digital Receipt management system. The store credentials for the Digital Receipt Service is maintained for upcoming receipt requests

7.7 API Certificate

To get access to the API the register partner / vendor is authorized to register and manage register identities by using a client certificate. The API Access Enrollment certificate can be distributed to the partner/vendor by the following methods:

1. Smartcard for HSM usage, status: transport revoked
2. USB, for files storage, status: transport revoked
3. HSM, by secure enrollment
4. Client CSR driven creation using Infrasec install client. Predefined offline secrets required that is generated during register enrollment.
5. RA Download, for file storage, two factor: offline secrets and mobile sms otp code, status transport revoked
6. API enrollment request may include section to create and POS Register (X.509) certificate. If the enrollment LRA Administrator Identity executing the enrollment request to create a certificate possess the "LRA (API) Extract Enroll Certificate" privilege the created certificate is also returned in the enrollment request.

8 Enrollment Information

The sections below describe the information for the request and response.

8.1 Information sections

The CCU enrollment request content is described below. There are 5 logical sections:

1. Heading for identifying the request
2. Action type of request
3. Tree structure and organizational information
4. Store information
5. POS information
6. Enrolled Services Information

8.2 Enrollment action types

The CCU enrollment request can be of one of the following action types:

1. NEW – new POS, adds a new Branch/Chain/Store/Register for accessing CCU slot or other service.
2. CLOSE – close a Register with CCU slot and or related services.
3. OPEN – reopen an already close and registered Register for CCU slot or other related services.
4. CHANGE – change Branch/Chain/Store/Register with service-related information.
5. STATUS – get status information for a tree object node, Path structural node or individual client
6. LIST – list information of a tree object node, Path structural node or individual client

The CCU enrollment request content is described below. There are 6 logical sections:

8.3 Tree structure prefix

In order to create a unique register identity, the nodes in the tree structure is bound to a unique set of characters we call prefix. The prefix is built as a concatenation of the superior node's prefixes. Thus, if the partner holds a partner authority code of AC01 and the nodes for chain and store holds RE and
1234567 and counter numbers are 1 and 2 the store registers would be named by the components AC01 RE 1234567 nnn where the register identities would be named AC01RE1234567001, AC01RE1234567002.

Each node in the tree can contribute with a prefix code that is used when creating the unique register identity The following prefix parts of the enrollment request can be used to create the register identities:

1. Partner Code, Mandatory
2. POS Authority Code, Mandatory
3. Branch Code
4. Chain Code
5. Store Id, Mandatory
6. Counter number

The number of characters each part in the prefix-code occupies and what prefix parts are used is defined by the agreement between the partner/vendor and Infrasec.

8.4 Status and List Path option

When executing a request to list or extract information from IDM the STATUS and LIST actions are used. The STATUS call can be used to identify a specific POS by supplying the RegisterID in the RegisterInfo section. Both the STATUS and LIST can be used to address a specific node in the tree by giving the tree path to the node to investigate. The List action will return information about the node and the immediate tree-child's of the node.

8.5 CCU Enrollment Request Information

The information in this section describes the various part of the enrollment request

8.5.1 Path or RegisterID

When the enrollment request is used to investigate information in the IDM Tree Structure using Status and List action, this can be done by either supplying the path to the node in the IDM Tree that shall investigated or by supplying a RegisterID of the POS Register that shall be investigated. The Path can be used when executing a List or Status action. The RegisterID cannot be used for the List action.

8.5.1.1 Path

The Path identifies a node or object in the IDM tree that shall be investigated. The Status and List action accepts the Path to an object that shall be investigated.

Setting the Path on a List to an empty string "" means that the node holding the LRA Administrator API client executing the request is used as the node for the Path to be listed.

When listing a node is a tree node that contain subordinate objects (Childs) to the node (Path) that is investigated, the child's are returned in a list where each child will be returned with a set of information and one attribute is the child's Path. The child's Path can then be used in a subsequent List or Status request.

8.5.1.2 RegisterID

The enrollment request holding a RegisterID identifies a register that shall be investigated. All request actions but the List action accepts the RegisterID as an identifier. The RegisterID is created on a successful New action. After this the RegisterID can be used on all Actions to manage the state and information of the register and business structure it belongs to.

8.5.2 Audit Identification

8.5.2.1 ApplicationID

(Optional)
A textual information that the partner use to identify the request application and is returned by the API on successfully executed response. Helps tracking information in audit logs

8.5.2.2 RequestID

(Optional)
A textual information that the partner use to identify the request message and is returned by the API on successfully executed response. Helps tracking information in audit logs

8.5.3 PartnerAuthority

8.5.3.1 PartnerName

(Mandatory)
Partner name identifies the partner in the partner tree, such as "Acme AB"

8.5.3.2 PartnerCode

(Mandatory)
Is created by Infrasec as the unique namespace for the partner, such as "AC".

8.5.3.3 POSAuthorityCode

(Mandatory)

POS Authority Code is created by Infrasec or partner. It describes the type of register ID structure.

8.5.4 *OrganizationBranch*

8.5.4.1 BranchName
(Optional)
Identifies a branch or division in the partner tree, such as "Restaurants"

8.5.4.2 BranchCode
(Optional)
Is created by partner unique for the branch node in the partner tree, such as "" (empty), "1" or "RE".

8.5.5 **OrganizationChain**

8.5.5.1 ChainName
(Optional)
Identifies a chain of stores in the partner tree, such as "Good Food Enterprise AB"

8.5.5.2 ChainCode
(Optional)
Is created by partner and is unique for the chain node in the partner tree, such as "" (empty), "GF", "1234" or the organization number "1212121212".

8.5.6 **StoreInfo**

8.5.6.1 StoreID
(Mandatory: new)
Must be a unique identity within the POS Vendors CCU slot tree, identifies the "store" that uses the POS device, such as "1", "1330" or the organization number "1212121212"

8.5.6.2 StoreName
(Mandatory: new)
The store name in the partner tree, such as "Good Food Restaurant One"

8.5.6.3 Email
(Optional)
Email address to responsible person at or for the store, may be a partner email

8.5.6.4 Cellphone
(Optional)
Cellphone where to send SMS information for the register, may be a partner cellphone

8.5.6.5 Address
(Mandatory: new)
Street address of the store.

8.5.6.6 Zipcode
(Mandatory: new)
Zipcode of the store.

8.5.6.7 City
(Mandatory: new)
City of the store.

8.5.7 **CompanyInfo**

8.5.7.1 OrganizationNumber
(Mandatory: new)
The Swedish Tax Authority registered organization number for the company owning the POS.

8.5.7.2 Company

(Mandatory: new)
Name of the company.

8.5.7.3 Address

(Mandatory: new)
Street address of the company.

8.5.7.4 Zipcode

(Mandatory: new)
Zipcode of the company.

8.5.7.5 City

(Mandatory: new)
City of the company.

8.5.8 RegisterInfo

8.5.8.1 RegisterID

(Mandatory: change, open, close, status)
The register identity is the POS identity that is used for the POS receipts, journals etc. The identity is communicated to the Tax Authority and it identifies a unique POS register. When registerid is supplied it must match the structure that is handled by the partner settings for the partner in the system. If the POS identity not computes the API call will fail.

8.5.8.2 RegisterMake

(Mandatory: new)
The make of the register, as stated in the self assessment with the Tax Authority "Självdeklaration"

8.5.8.3 RegisterModel

(Mandatory: new)
The model of the register, as stated in the self assessment with the Tax Authority "Självdeklaration"

8.5.8.4 LocalAlias

(Optional)
Local alias of the register, this may be a historic name of the register or used locally to bind the CCU Register Identity for local administration purposes.

8.5.8.5 CounterNumber

(Optional)
The number of this register at the store, part of POS register identity together with partner, branch and chain codes together with StoreID. If empty next available store will be deduced by the Enrollment Process.

8.5.8.6 Address

(Mandatory: new)
Street ddress of the register location.

8.5.8.7 Zipcode

(Mandatory: new)
Zipcode of the register location.

8.5.8.8 City

(Mandatory: new)
City of the register location.

8.5.9 *JournalLocation*

The physical location of the Journal memory.

8.5.9.1 Company

(Mandatory: new)
Name of the company as registered at Swedish "Bolagsverket" where Journal is available.

8.5.9.2 Address

(Mandatory: new)
Street address where Journal is available.

8.5.9.3 Zipcode

(Mandatory: new)
Zipcode where Journal is available.

8.5.9.4 City

(Mandatory: new)
City where Journal is available.

8.5.10 *OperationLocation*

The location of the running software.

8.5.10.1 Company

(Mandatory: new)
Name of the company as registered at Swedish "Bolagsverket" where Operation Center for register execution is located.

8.5.10.2 Address

(Mandatory: new)
Street address where Operation Center for register execution is located.

8.5.10.3 Zipcode

(Mandatory: new)
Zipcode where Operation Center for register execution is located.

8.5.10.4 City

(Mandatory: new)
City where Operation Center for register execution is located.

8.5.11 *PcxService*

List of PosConneXion Services to enroll for

8.5.11.1 CCU

8.5.11.1.1 Enable
- Yes, Enroll register for CCU Access
- No, Do not enroll register for CCU Access

8.5.11.2 Swish

8.5.11.2.1 Enable
- Yes, Enroll register for integrated Swish payments

- No, Do not enroll register for integrated Swish payments

8.5.11.2.2 SwishNr

Swish nr where store is accepting swish payments (123xxx)

8.5.11.2.3 SwishType

SWEDBANK
GETSWISH

8.5.12 *Certificate*

Prepare credentials for or execute complete enrollment of POS or Agent certificate. The enrollment executed will follow the configured distribution model configured. The distribution model defines how partner certificates are managed and is a step that is selected in cooperation with Infrasec when partner is configured or managed in IDM.

8.5.12.1 Method

Current value is: ENROLL

8.5.12.2 Email

Email address to responsible partner administrator or for distribution method selected recipient for certificate credentials

8.5.12.3 Cellphone

Cellphone to responsible partner administrator or for distribution method selected recipient for certificate credentials

8.6 CCU Enrollment Response Information

The information in this section describes the various part of the enrollment request

Example successful response:

```
{"success":true,
"message":"",
"IdmResponse":{
    "ResponseCode":"0",
    "ResponseMessage":"Success",
    "ApplicationID":"Acme EnrollApp 1",
    "RequestID":"12345",
    "Action":"NEW",
    "RegisterID":"AC01121212121201",
    "CCUID":"ISCCUSE0000000011"}}
```

Example non successful response:

```
{"success":false,
"message":"RegisterID AC01121212121201 already exists",
"IdmResponse":{
    "ResponseCode":86,
    "ResponseMessage":"Request Failed",
    "ResponseReason":"RegisterID AC01121212121201 already exists"}}
```

8.6.1   *Audit Identification*

8.6.1.1 success

Boolean response that inform of the completion of the request
1. true, the execution was considered to complete successfully
2. false, the execution was considered to not complete successfully

8.6.1.2 message

Information message why the request was not considered successful

8.6.2   *IdmResponse*

8.6.2.1 ResponseReason

May contain extra information about the error.

8.6.2.2 ApplicationID

As from request textual information that partner use to identify the request application and is returned by the API on successfully executed response. Helps tracking information in audit logs

8.6.2.3 RequestID

As from request textual information that partner use to identify the request message and is returned by the API on successfully executed response. Helps tracking information in audit logs

8.6.2.4 Action

Requested action as from the request; new, change, close, open, status.

8.6.2.5 RegisterID

The register identity for the "Tillverkningsnummer/kassabeteckning", the unique identity of the cash register.

8.6.2.6 CCUID

The identity of the CCU that is preallocated for this register. Note that this may change at will on service request or on other requirements stated by Infrasec. On such change Infrasec will inform the Tax Authority that the CCU has been changed.

8.6.2.7 Active

If Status was reqested then either 0 or 1 will be returned. 0 for inactive, 1 for active.

9 Enrollment API JSON Example

9.1 New, Enroll CCU

Request:

```json
{"IdmRequest": {
    "ApplicationID":"Acme EnrollApp 1",
    "RequestID": "123456",
    "EnrollData": {
      "Action": "NEW",
      "PartnerAuthority": {
        "PartnerName": "Acme AB",
        "PartnerCode": "AC",
        "POSAuthorityCode": "01"
      },
      "OrganizationBranch": {
        "BranchName": "",
        "BranchCode": ""
      },
      "OrganizationChain": {
        "ChainName": "",
        "ChainCode": ""
      },
      "StoreInfo": {
        "StoreID": "1212121212",
        "StoreName": "Wiles",
        "Email": "w@wiles.se",
        "Cellphone": "+46123456789",
        "Address": "Testgatan 1",
        "Zipcode": "123 45",
        "City": "Teststad"
      },
      "CompanyInfo": {
        "OrganizationNumber": "121212-1212",
        "Company": "Wiles Handel AB",
        "Address": "Testgatan 1",
        "Zipcode": "123 45",
        "City": "Teststaden"
      },
      "RegisterInfo": {
        "RegisterID": "",
        "RegisterMake": "Acme 1",
        "RegisterModel": "x.x.x.x",
        "LocalAlias": "Kassa 1",
        "CounterNumber": "1",
        "Address": "Testgatan 1",
        "Zipcode": "123 45",
        "City": "Teststaden"
      },
      "JournalLocation": {
        "Company": "Acme AB",
        "Address": "Testgatan 1",
        "Zipcode": "123 45",
        "City": "Teststaden"
```

```
      },
      "OperationLocation": {
        "Company": "Acme AB",
        "Address": "Testgatan 1",
        "Zipcode": "123 45",
        "City": "Teststaden"
      },
      "PcxService": {
        "CCU": {
          "Enable": "Yes"
        },
        "Swish": {
          "Enable": "No",
          "SwishNr": "",
          "SwishType": "2"
        }
}}}}
```

Response:

```
{"IdmResponse": {
    "ResponseCode": "0",
    "ResponseMessage": "Success",
    "ApplicationID": "Acme EnrollApp 1",
    "RequestID": "123456",
    "Action": "NEW",
    "RegisterID": "AC01121212121201",
    "CCUID": "ISCCUSE0000000011"
  }}
```

9.2 Close
Request:

```
{"IdmRequest": {
    "EnrollData": {
      "Action": "CLOSE",
      "RegisterInfo": {
        "RegisterID": "AC01121212121201"
      }}}}
```

Response:

```
{"IdmResponse": {
    "ResponseCode": "0",
    "ResponseMessage": "Success",
    "Action": "CLOSE",
    "RegisterID": "AC01121212121201",
    "CCUID": "ISCCUSE0000000011"
  }}
```

9.3 Open
Request:

```
{"IdmRequest": {
    "EnrollData": {
```

```
    "Action": "OPEN",
    "RegisterInfo": {
      "RegisterID": "AC01121212121201"
    }}}}
```

Response:

```
{"IdmResponse": {
   "ResponseCode": "0",
   "ResponseMessage": "Success",
   "ApplicationID": "Acme EnrollApp 1",
   "RequestID": "123456",
   "Action": "OPEN",
   "RegisterID": "AC01121212121201",
   "CCUID": "ISCCUSE0000000011"
  }}
```

9.4 Change

Request:

```
{"IdmRequest": {
   "ApplicationID":"Acme EnrollApp 1",
   "RequestID": "123456",
   "EnrollData": {
    "Action": "CHANGE",
    "PartnerAuthority": {
      "PartnerName": "Acme AB",
      "PartnerCode": "AC",
      "POSAuthorityCode": "01"
    },
    "OrganizationBranch": {
      "BranchName": "",
      "BranchCode": ""
    },
    "OrganizationChain": {
      "ChainName": "",
      "ChainCode": ""
    },
    "StoreInfo": {
      "StoreID": "1212121212",
      "StoreName": "Wiles",
      "Email": "w@wiles.se",
      "Cellphone": "+46123456789",
      "Address": "Testgatan 1",
      "Zipcode": "123 45",
      "City": "Teststad"
    },
    "CompanyInfo": {
      "OrganizationNumber": "121212-1212",
      "Company": "Wiles Handel AB",
      "Address": "Testgatan 1",
      "Zipcode": "123 45",
      "City": "Teststaden"
    },
    "RegisterInfo": {
      "RegisterID": "",
```

```
    "RegisterMake": "Acme 1",
    "RegisterModel": "x.x.x.x",
    "LocalAlias": "Kassa 1",
    "CounterNumber": "1",
    "Address": "Testgatan 1",
    "Zipcode": "123 45",
    "City": "Teststaden"
   },
   "JournalLocation": {
    "Company": "Acme AB",
    "Address": "Testgatan 1",
    "Zipcode": "123 45",
    "City": "Teststaden"
   },
   "OperationLocation": {
    "Company": "Acme AB",
    "Address": "Testgatan 1",
    "Zipcode": "123 45",
    "City": "Teststaden"
   },
   "PcxService": {
    "CCU": {
     "Enable": "Yes"
    },
    "Swish": {
     "Enable": "No",
     "SwishNr": "",
     "SwishType": "2"
    },
    "Sparakvittot": {
     "Enable": "No",
     "SparakvittotAccount": "",
     "SparakvittotStoreid": "",
     "SparakvittotUsername": "",
     "SparakvittotPassword": ""
    }}}}}
```

Response:

```
{"IdmResponse": {
   "ResponseCode": "0",
   "ResponseMessage": "Success",
   "ApplicationID": "Acme EnrollApp 1",
   "RequestID": "123456",
   "Action": "CHANGE",
   "RegisterID": "AC01121212121201",
   "CCUID": "ISCCUSE0000000011"
 }}
```

9.5 Status on RegisterID

Request:

```
{"IdmRequest": {
   "EnrollData": {
    "Action": "STATUS",
```

```
    "RegisterInfo": {
      "RegisterID": "AC01121212121201"
    }}}}
```

Response:
```
{"IdmResponse": {
   "ResponseCode": "0",
   "ResponseMessage": "Success",
   "Action": "STATUS",
   "CCUID": "ISCCUSE0000000011",
   "Active": "1",
   "LoginCount": "1",
   "LastLogin": "2016-02-02 18:27:12",
   "Info": {
     "PartnerAuthority": {
       "PartnerName": "Acme AB",
       "PartnerCode": "AC",
       "POSAuthorityCode": "01"
     },
     "StoreInfo": {
       "StoreID": "1212121212",
       "StoreName": "Wiles",
       "Email": "w@wiles.se",
       "Cellphone": "+46123456789",
       "Address": "Testgatan 1",
       "Zipcode": "123 45",
       "City": "Teststad"
     },
     "CompanyInfo": {
       "OrganizationNumber": "121212-1212",
       "Company": "Wiles Handel AB",
       "Address": "Testgatan 1",
       "Zipcode": "123 45",
       "City": "Teststaden"
     },
     "RegisterInfo": {
       "RegisterID": "AC01121212121201",
       "RegisterMake": "Acme 1",
       "RegisterModel": "x.x.x.x",
       "LocalAlias": "Kassa 1",
       "CounterNumber": "1",
       "Address": "Testgatan 1",
       "Zipcode": "123 45",
       "City": "Teststaden"
     },
     "JournalLocation": {
       "Company": "Acme AB",
       "Address": "Testgatan 1",
       "Zipcode": "123 45",
       "City": "Teststaden"
     },
     "OperationLocation": {
       "Company": "Acme AB",
       "Address": "Testgatan 1",
       "Zipcode": "123 45",
       "City": "Teststaden"}}}}
```

9.6 Status on Path

Request:

```
{
  "IdmRequest": {
    "ApplicationID": "ACME Verify POS",
    "RequestID": "329229",
    "EnrollData": {
      "Action": "STATUS",
      "PartnerAuthority": {
        "PartnerName": "Acme AB",
        "PartnerCode": "AA",
        "POSAuthorityCode": "01"
      },
      "Path": "\/Acme AB\/Acme 1"
    }
  }
}
```

Response:

```
{
  "IdmResponse": {
    "ResponseCode": "0",
    "ResponseMessage": "Success",
    "total": 1,
    "Action": "STATUS",
    "Path": "\/Acme AB\/Acme 1",
    "rows": {
      "PartnerAuthority": {
        "PartnerName": "Acme AB",
        "PartnerCode": "AA",
        "POSAuthorityCode": "01"
      },
      "PartnerPOSAuthorityInfo": {
        "Path": "\/Acme AB\/Acme 1",
        "Prefix": "AA01",
        "PartnerPOSAuthorityName": "Acme 1",
        "PartnerPOSAuthorityCode": "01",
        "Email": "",
        "Cellphone": "",
        "ExpireCode": "0",
        "ExpireDate": null,
        "ActiveDate": null,
        "LocationList": {
          "PartnerPOSAuthorityLocation": {
            "Company": "",
            "Address": "",
            "Zipcode": "",
            "City": ""
          }
        },
        "MemberList": {
          "PrivateMember": {
            "MemberID": "",
            "UserID": "",
```

```
        "Password": "",
        "Statistics": null
      }
     }
    }
   }
  }
}
```

9.7   List on Path

Request:
```
{
 "IdmRequest": {
  "ApplicationID": "ACME Verify POS",
  "RequestID": "207281",
  "EnrollData": {
   "Action": "LIST",
   "PartnerAuthority": {
    "PartnerName": "Acme AB",
    "PartnerCode": "AA",
    "POSAuthorityCode": "01"
   },
   "Path": "\/Acme AB\/Acme 1"
  }
 }
}
```

Response:
```
{
 "IdmResponse": {
  "ResponseCode": "0",
  "ResponseMessage": "Success",
  "total": 5,
  "Action": "LIST",
  "Path": "\/Acme AB\/Acme 1",
  "rows": [{
   "Name": "Electronics",
   "Path": "\/Acme AB\/Acme 1\/Electronics",
   "Type": "12",
   "Info": "Branch",
   "OrgNr": "",
   "Prefix": "AA01EL",
   "Grouping": "EL",
   "Leaf": "false",
   "ExpireCode": "0",
   "ExpireDate": null,
   "ActiveDate": null,
   "ChildCount": "1"
  }, {
   "Name": "Food & Beverage",
   "Path": "\/Acme AB\/Acme 1\/Food & Beverage",
   "Type": "12",
   "Info": "Branch",
   "OrgNr": "",
   "Prefix": "AA01FB",
   "Grouping": "FB",
```

"Leaf": "false",
"ExpireCode": "0",
"ExpireDate": null,
"ActiveDate": null,
"ChildCount": "2"
}....

10 Enrollment API XML Example

10.1 New, enroll register to CCU Service

Request:

```
<IdmRequest>
<ApplicationID>Acme Enrol lApp 1</ApplicationID>
<RequestID>910324</RequestID>
<EnrollData>
  <Action>NEW</Action>
  <PartnerAuthority>
   <PartnerName>Acme AB</PartnerName>
   <PartnerCode>AC</PartnerCode>
   <POSAuthorityCode>1</POSAuthorityCode>
  </PartnerAuthority>
  <OrganizationBranch>
   <BranchName></BranchName>
   <BranchCode></BranchCode>
  </OrganizationBranch>
  <OrganizationChain>
   <ChainName><![CDATA[Selling Things]]></ChainName>
   <ChainCode>ST01</ChainCode>
  </OrganizationChain>
  <StoreInfo>
   <StoreID>00001</StoreID>
   <StoreName><![CDATA[Selling Things With ÅÄÖ & åäö and é]]></StoreName>
   <Address><![CDATA[Test Street 1]]></Address>
   <Zipcode>111 11</Zipcode>
   <City><![CDATA[Test City]]></City>
  </StoreInfo>
  <CompanyInfo>
   <OrganizationNumber>111111-1111</OrganizationNumber>
   <Company><![CDATA[Selling Things AB]]></Company>
   <Address><![CDATA[Test Street 1]]></Address>
   <Zipcode>111 11</Zipcode>
   <City><![CDATA[Test City]]></City>
  </CompanyInfo>
  <RegisterInfo>
   <RegisterID></RegisterID>
   <RegisterMake><![CDATA[Acme 1]]></RegisterMake>
   <RegisterModel><![CDATA[x.x.x.x]]></RegisterModel>
   <LocalAlias><![CDATA[]]></LocalAlias>
   <CounterNumber></CounterNumber>
   <Address><![CDATA[Test Street 1]]></Address>
   <Zipcode>111 11</Zipcode>
   <City><![CDATA[Test City]]></City>
  </RegisterInfo>
  <JournalLocation>
   <Company><![CDATA[Selling Things AB]]></Company>
   <Address><![CDATA[Test Street 1]]></Address>
   <Zipcode>111 11</Zipcode>
   <City><![CDATA[Test City]]></City>
  </JournalLocation>
  <OperationLocation>
```

```xml
    <Company><![CDATA[Selling Things AB]]></Company>
    <Address><![CDATA[Test Street 1]]></Address>
    <Zipcode>111 11</Zipcode>
    <City><![CDATA[Test City]]></City>
  </OperationLocation>
  <Certificate>
    <Method>ENROLL</Method>
    <Email>user@acmeacme.se</Email>
    <Cellphone>+4612345678</Cellphone>
  </Certificate>
  <PcxService>
   <CCU>
     <Enable>Yes</Enable>
   </CCU>
   <Swish>
     <Enable>No</Enable>
     <SwishNr></SwishNr>
     <SwishType></SwishType>
   </Swish>
   <Sparakvittot>
     <Enable>No</Enable>
     <SparakvittotAccount></SparakvittotAccount>
     <SparakvittotStoreid></SparakvittotStoreid>
     <SparakvittotUsername></SparakvittotUsername>
     <SparakvittotPassword></SparakvittotPassword>
   </Sparakvittot>
  </PcxService>
</EnrollData>
</IdmRequest>
```

Response:

```xml
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <Action>NEW</Action>
  <RegisterID>AC01121212121201</RegisterID>
  <CCUID>ISCCUSE0000000011</CCUID>
</IdmResponse>
```

10.1 New, enroll register to CCU Service with Certificate Generation

Request:

```xml
<IdmRequest>
<ApplicationID>Acme Enrol lApp 1</ApplicationID>
<RequestID>910324</RequestID>
<EnrollData>
  <Action>NEW</Action>
  <PartnerAuthority>
    <PartnerName>Acme AB</PartnerName>
    <PartnerCode>AC</PartnerCode>
    <POSAuthorityCode>1</POSAuthorityCode>
  </PartnerAuthority>
```

```xml
<OrganizationBranch>
  <BranchName></BranchName>
  <BranchCode></BranchCode>
</OrganizationBranch>
<OrganizationChain>
  <ChainName><![CDATA[Selling Things]]></ChainName>
  <ChainCode>ST01</ChainCode>
</OrganizationChain>
<StoreInfo>
  <StoreID>00001</StoreID>
  <StoreName><![CDATA[Selling Things With ÅÄÖ & åäö and é]]></StoreName>
  <Address><![CDATA[Test Street 1]]></Address>
  <Zipcode>111 11</Zipcode>
  <City><![CDATA[Test City]]></City>
</StoreInfo>
<CompanyInfo>
  <OrganizationNumber>111111-1111</OrganizationNumber>
  <Company><![CDATA[Selling Things AB]]></Company>
  <Address><![CDATA[Test Street 1]]></Address>
  <Zipcode>111 11</Zipcode>
  <City><![CDATA[Test City]]></City>
</CompanyInfo>
<RegisterInfo>
  <RegisterID></RegisterID>
  <RegisterMake><![CDATA[Acme 1]]></RegisterMake>
  <RegisterModel><![CDATA[x.x.x.x]]></RegisterModel>
  <LocalAlias><![CDATA[]]></LocalAlias>
  <CounterNumber></CounterNumber>
  <Address><![CDATA[Test Street 1]]></Address>
  <Zipcode>111 11</Zipcode>
  <City><![CDATA[Test City]]></City>
</RegisterInfo>
<JournalLocation>
  <Company><![CDATA[Selling Things AB]]></Company>
  <Address><![CDATA[Test Street 1]]></Address>
  <Zipcode>111 11</Zipcode>
  <City><![CDATA[Test City]]></City>
</JournalLocation>
<OperationLocation>
  <Company><![CDATA[Selling Things AB]]></Company>
  <Address><![CDATA[Test Street 1]]></Address>
  <Zipcode>111 11</Zipcode>
  <City><![CDATA[Test City]]></City>
</OperationLocation>
<Certificate>
  <Method>ENROLL</Method>
  <Email>user@acmeacme.se</Email>
  <Cellphone>+4612345678</Cellphone>
</Certificate>
<PcxService>
 <CCU>
  <Enable>Yes</Enable>
 </CCU>
 <Swish>
  <Enable>No</Enable>
  <SwishNr></SwishNr>
```

```
    <SwishType></SwishType>
   </Swish>
   <Sparakvittot>
    <Enable>No</Enable>
    <SparakvittotAccount></SparakvittotAccount>
    <SparakvittotStoreid></SparakvittotStoreid>
    <SparakvittotUsername></SparakvittotUsername>
    <SparakvittotPassword></SparakvittotPassword>
   </Sparakvittot>
  </PcxService>
 </EnrollData>
</IdmRequest>
```

Response:

```
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <Action>NEW</Action>
  <RegisterID>AC01121212121201</RegisterID>
  <CCUID>ISCCUSE0000000011</CCUID>
</IdmResponse>
```

10.2 Close

Request:

```
<IdmRequest>
  <EnrollData>
    <Action>CLOSE</Action>
    <RegisterInfo>
      <RegisterID>AC01121212121201</RegisterID>
    </RegisterInfo>
  </EnrollData>
</IdmRequest>
```

Response:

```
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <Action>CLOSE</Action>
  <RegisterID>AC01121212121201</RegisterID>
  <CCUID>ISCCUSE0000000011</CCUID>
</IdmResponse>
```

10.3 Open

Request:

```
<IdmRequest>
  <EnrollData>
    <Action>OPEN</Action>
    <RegisterInfo>
```

```
    <RegisterID>AC01121212121201</RegisterID>
   </RegisterInfo>
  </EnrollData>
</IdmRequest>
```

Response:

```
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <Action>OPEN</Action>
  <RegisterID>AC01121212121201</RegisterID>
  <CCUID>ISCCUSE0000000011</CCUID>
</IdmResponse>
```

10.4 Change

Request:

```
<IdmRequest>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <EnrollData>
   <Action>CHANGE</Action>
   <PartnerAuthority>
    <PartnerName>Acme AB</PartnerName>
    <PartnerCode>AC</PartnerCode>
    <POSAuthorityCode>1</POSAuthorityCode>
   </PartnerAuthority>
   <OrganizationBranch>
    <BranchName></BranchName>
    <BranchCode></BranchCode>
   </OrganizationBranch>
   <OrganizationChain>
    <ChainName></ChainName>
    <ChainCode></ChainCode>
   </OrganizationChain>
   <StoreInfo>
    <StoreID>1212121212</StoreID>
    <StoreName>Wiles Butik</StoreName>
    <Address>Testgatan 1</Address>
    <Zipcode>123 45</Zipcode>
    <City>Teststaden</City>
   </StoreInfo>
   <CompanyInfo>
    <OrganizationNumber>121212-1212</OrganizationNumber>
    <Company>Wiles Handel AB</Company>
    <Address>Testgatan 1</Address>
    <Zipcode>123 45</Zipcode>
    <City>Teststaden</City>
   </CompanyInfo>
   <RegisterInfo>
    <RegisterID></RegisterID>
    <RegisterMake>Acme 1</RegisterMake>
    <RegisterModel>x.x.x.x</RegisterModel>
    <LocalAlias>Kassa</LocalAlias>
    <CounterNumber></CounterNumber>
```

```xml
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </RegisterInfo>
    <JournalLocation>
      <Company>Acme AB</Company>
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </JournalLocation>
    <OperationLocation>
      <RegisterID>Acme AB</RegisterID>
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </OperationLocation>
    <PcxService>
      <CCU>
        <Enable>Yes</Enable>
      </CCU>
      <Swish>
        <Enable>No</Enable>
        <SwishNr>123999999</SwishNr>
        <SwishType>2</SwishType>
      </Swish>
      <Sparakvittot>
        <Enable>No</Enable>
        <SparakvittotAccount></SparakvittotAccount>
        <SparakvittotStoreid></SparakvittotStoreid>
        <SparakvittotUsername></SparakvittotUsername>
        <SparakvittotPassword></SparakvittotPassword>
      </Sparakvittot>
    </PcxService>
  </EnrollData>
</IdmRequest>
```

Response:

```xml
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <Action>CHANGE</Action>
  <RegisterID>AC01121212121201</RegisterID>
  <CCUID>ISCCUSE0000000011</CCUID>
</IdmResponse>
```

10.5 Status

Request:

```xml
<IdmRequest>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <EnrollData>
    <Action>STATUS</Action>
    <RegisterInfo>
      <RegisterID>AC01121212121201</RegisterID>
    </RegisterInfo>
  </EnrollData>
</IdmRequest>
```

Response:

```xml
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <Action>STATUS</Action>
  <CCUID>ISCCUSE0000000011</CCUID>
  <Active>1</Active>
  <LoginCount>1</LoginCount>
  <LastLogin>2016-02-02 18:27:12</LastLogin>
  <Info>
    <PartnerAuthority>
      <PartnerName>Acme AB</PartnerName>
      <PartnerCode>AC</PartnerCode>
      <POSAuthorityCode>1</POSAuthorityCode>
    </PartnerAuthority>
    <StoreInfo>
      <StoreID>1212121212</StoreID>
      <StoreName>Wiles Butik</StoreName>
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </StoreInfo>
    <CompanyInfo>
      <OrganizationNumber>121212-1212</OrganizationNumber>
      <Company>Wiles Handel AB</Company>
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </CompanyInfo>
    <RegisterInfo>
      <RegisterID>AC01121212121201</RegisterID>
      <RegisterMake>Acme 1</RegisterMake>
      <RegisterModel>x.x.x.x</RegisterModel>
      <LocalAlias>Kassa 1</LocalAlias>
      <CounterNumber>1</CounterNumber>
      <Address>Testgatan 1</Address>
      <Zipcode>123 45</Zipcode>
      <City>Teststaden</City>
    </RegisterInfo>
    <JournalLocation>
      <Company>Acme AB</Company>
```

```
    <Address>Testgatan 1</Address>
    <Zipcode>123 45</Zipcode>
    <City>Teststaden</City>
   </JournalLocation>
   <OperationLocation>
    <RegisterID>Acme AB</RegisterID>
    <Address>Testgatan 1</Address>
    <Zipcode>123 45</Zipcode>
    <City>Teststaden</City>
   </OperationLocation>
  </Info>
</IdmResponse>
```

11 XML Path List Examples

11.1 List on empty Path

When the request List on a empty path, the root path of where the Enrollment client is located in the IDM Tree. In this case the is the base for where the client is "Acme Enrollmenmt Verify API" is located, "/Acme AB"

Request:
```
<IdmRequest>
  <ApplicationID>Acme EnrollApp 1</ApplicationID>
  <RequestID>123456</RequestID>
  <EnrollData>
   <Action>LIST</Action>
   <PartnerAuthority>
    <PartnerName>Acme AB</PartnerName>
    <PartnerCode>AC</PartnerCode>
    <POSAuthorityCode>1</POSAuthorityCode>
   </PartnerAuthority>
   <Path><![CDATA[]]></Path>
   </EnrollData>
  </IdmRequest>
```

Response:
```
<?xml version="1.0"?>
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <total>2</total>
  <Action>LIST</Action>
  <Path>/Acme AB</Path>
  <rows>
   <item0>
    <Name>Acme 1</Name>
    <Path>/Acme AB/Acme 1</Path>
    <Type>33</Type>
    <Info>Partner POS Authority</Info>
    <OrgNr/>
    <Prefix>AA01</Prefix>
    <Grouping>01</Grouping>
    <Leaf>false</Leaf>
```

```xml
    <ExpireCode>0</ExpireCode>
    <ExpireDate/>
    <ActiveDate/>
    <ChildCount>5</ChildCount>
   </item0>
   <item1>
    <Name>Acme Enrollmenmt Verify API</Name>
    <Path>/Acme AB/Acme Enrollmenmt Verify API</Path>
    <Type>8</Type>
    <Info>Local Administrator</Info>
    <OrgNr>0000000000</OrgNr>
    <Prefix/>
    <Grouping/>
    <Leaf>true</Leaf>
    <Enabled>1</Enabled>
    <LastLogin>2021-11-08 17:34:24</LastLogin>
   </item1>
  </rows>
</IdmResponse>
```

11.2 List on named Path

This request List a named path as the returned path in item0 above, In this case the is the path is the base for where the list is executed, "/Acme AB/Acme 1"

Request:
```xml
<IdmRequest>
  <ApplicationID>ACME Verify POS</ApplicationID>
  <RequestID>556399</RequestID>
  <EnrollData>
   <Action>LIST</Action>
   <PartnerAuthority>
    <PartnerName>Acme AB</PartnerName>
    <PartnerCode>AA</PartnerCode>
    <POSAuthorityCode>01</POSAuthorityCode>
   </PartnerAuthority>
   <Path><![CDATA[/Acme AB/Acme 1]]></Path>
   </EnrollData>
  </IdmRequest>
```

Response:
```xml
"<?xml version="1.0"?>
<IdmResponse>
  <ResponseCode>0</ResponseCode>
  <ResponseMessage>Success</ResponseMessage>
  <total>5</total>
  <Action>LIST</Action>
  <Path>/Acme AB/Acme 1</Path>
  <rows>
   <item0>
    <Name>Electronics</Name>
    <Path>/Acme AB/Acme 1/Electronics</Path>
    <Type>12</Type>
    <Info>Branch</Info>
    <OrgNr/>
    <Prefix>AA01EL</Prefix>
```

<Grouping>EL</Grouping>
<Leaf>false</Leaf>
<ExpireCode>0</ExpireCode>
<ExpireDate/>
<ActiveDate/>
<ChildCount>1</ChildCount>
</item0>
<item1>
<Name>Food & Beverage</Name>
<Path>/Acme AB/Acme 1/Food & Beverage</Path>
<Type>12</Type>
<Info>Branch</Info>
<OrgNr/>
<Prefix>AA01FB</Prefix>
<Grouping>FB</Grouping>
<Leaf>false</Leaf>
<ExpireCode>0</ExpireCode>
<ExpireDate/>
<ActiveDate/>
<ChildCount>2</ChildCount>
</item1>
<item2>
<Name>Men Shoes</Name>
<Path>/Acme AB/Acme 1/Men Shoes</Path>
<Type>12</Type>
<Info>Branch</Info>
<OrgNr/>
<Prefix>AA01SH</Prefix>
<Grouping>SH</Grouping>
<Leaf>false</Leaf>
<ExpireCode>0</ExpireCode>
<ExpireDate/>
<ActiveDate/>
<ChildCount>1</ChildCount>
</item2>
<item3>
<Name>POS A2</Name>
<Path>/Acme AB/Acme 1/POS A2</Path>
<Type>28</Type>
<Info>POS Service Asset</Info>
<OrgNr/>
<Prefix/>
<Grouping/>
<Leaf>true</Leaf>
<Enabled>1</Enabled>
<LastLogin/>
</item3>
<item4>
<Name>Women Shoes</Name>
<Path>/Acme AB/Acme 1/Women Shoes</Path>
<Type>12</Type>
<Info>Branch</Info>
<OrgNr/>
<Prefix>AA01WH</Prefix>
<Grouping>WH</Grouping>
<Leaf>false</Leaf>

        <ExpireCode>0</ExpireCode>
        <ExpireDate/>
        <ActiveDate/>
        <ChildCount>1</ChildCount>
      </item4>
    </rows>
  </IdmResponse>