

APIs by Example: Validation List APIs

[System i Network Programming Tips Newsletter](#)

System iNEWS Staff

Thu, 04/12/2001 (All day)

Since V4R1, IBM has offered validation list objects, which allow you to identify and authenticate users of your Web applications. One of the benefits of validation lists is that they aren't tied to AS/400 user profiles and passwords, so Web users don't have access to publicly authorized objects on your system. Validation lists aren't restricted to Web applications - any application can use them, and IBM provides several APIs for just that purpose.

Validation lists are discussed in more detail in an article at <http://www.as400network.com/article.cfm?ID=2057> (available to AS400 Network professional users only).

This installment of APIs by Example provides two sample RPG programs, service program CBX0031 and program CBX003T, which tests the APIs. CBX0031 appears in this newsletter; for space reasons, CBX003T is omitted, but you can download it from

<http://www.as400network.com/noderesources/code/clubtechcode/ValidationListAPIs.zip>.

The Validation List APIs used in program CBX0031 include the following:

QsyAddValidationLstEntry

Adds an entry to a validation list object. The entry is defined by an Entry ID (user name), Encrypt Data (user password), and Entry Data (user description).

QsyFindValidationLstEntry

Finds an entry in a validation list object. To retrieve the password, the system value RETSVRSEC must be set to 1 and the QsyEncryptData attribute of the entry should have been set to 1 when the entry was stored.

QsyChangeValidationLstEntry

Changes an entry in a validation list object. You can change the password and description values as well as attribute information.

QsyRemoveValidationLstEntry

Removes an entry from a validation list object.

QsyVerifyValidationLstEntry

Verifies the existence of the specified validation list entry and the validity of the specified password.

All of the above APIs are described in detail at

<http://publib.boulder.ibm.com/pubs/html/as400/v4r5/ic2924/index.htm?info/apis/sec6.htm>

Here's program CBX0031:

```

** Exported functions:
**
** VfyUsrPwd  Verifies the username and password          Return values:
**            against the specified validation            -1: Failure
**            list.                                       -2: Password error
**
** AddUsrPwd  Adds a username, a password and a          -1: Failure
**            description to the specified               -2: Password error
**            validation list.
**
** ChgUsrPwd  Changes the password of the speci-        -1: Failure
**            fied username in the specified            -1: Password error
**            validation list.

```

```

**
** GetUsrPwd Returns the password of the speci-      Blanks: Failure
**          fied username if found in the
**          specified validation list.
**
**
** RmvUsrPwd Removes the validation list entry      -1: Failure
**          of the specified username in the
**          specified validation list.
**
** Compile options:
**   CRTRPGMOD      CBX003
**   CRTSRVPGM      CBX003
**
**-- Header specifications: -----**
H NoMain Option( *SrcStmt )
**-- Entry Id: -----**
D EntId          Ds
D  EiDtaLen      10i 0
D  EiCcsId       10i 0 Inz( 65535 )
D  EiDta         10a
**-- Encryption data: -----**
D EncDta        Ds
D  EcDtaLen      10i 0
D  EcCcsId       10i 0 Inz( 65535 )
D  EcDta         10a
**-- Entry data: -----**
D EntDta        Ds
D  EdDtaLen      10i 0
D  EdCcsID       10i 0 Inz( 65535 )
D  EdDta         50a
**-- Attribute data: -----**
D AtrInf        Ds
D  AiNbrAtr      10i 0 Inz( 1 )
D  AiAlign       12a
** Qsy_Attr_Descr_T
D  AdAttrLoc      10i 0 Inz( 0 )
D  AdAttrTyp      10i 0 Inz( 0 )
D  AdRes_1        8a  Inz( *Allx'00' )
D  AdAttrId_p     *
D  AdOthDsc       32a  Inz( *Allx'00' )
** Attr_Data_Info
** Qsy_In_VLDL_T
D  VldLstAtr      96a
D  AvAttrCcsid    10i 0 Overlay( VldLstAtr: 1 )
D                      Inz( -1 )
D  AvAttrLen      10i 0 Overlay( VldLstAtr: 5 )
D                      Inz( 1 )
D  AvRes_1        8a  Overlay( VldLstAtr: 9 )
D                      Inz( *Allx'00' )
D  AvAttrVal_p    *  Overlay( VldLstAtr: 17 )
D                      Inz( %Addr( AvAttrVal ) )
D  VaInOther      96a Overlay( VldLstAtr: 1 )
D                      64a Overlay( VaInOther: 33 )
D                      Inz( *Allx'00' )
D  AdOtherData    32a Inz( *Allx'00' )
**
D AvAttrVal      s          1a  Inz( '1' )
**-- Return data: -----**
D RtnDta        Ds
** Qsy_Entry_ID_Info_T
D  RiDtaLen      10i 0
D  RiCcsId       10i 0
D  RiDta         100a
** Qsy_Entry_Encr_Data_Info_T
D  RcDtaLen      10i 0
D  RcCcsId       10i 0
D  RcDta         600a
** Qsy_Entry_Data_Info_T
D  RdDtaLen      10i 0
D  RdCcsID       10i 0
D  RdDta         1000a
D                      4a
D  RaAttrPtr     *
**-- Verify user password: -----**
D VfyUsrPwd      Pr          10i 0

```

```

D PxVldLst          10a  Const
D PxVldLstLib       10a  Const
D PxUsrId           10a  Const
D PxUsrPwd          10a  Const
**-- Add user password: -----**
D AddUsrPwd      Pr          10i 0
D PxVldLst          10a  Const
D PxVldLstLib       10a  Const
D PxUsrId           10a  Const
D PxUsrPwd          10a  Const
D PxUsrDsc          50a  Const
**-- Get user password: -----**
D GetUsrPwd      Pr          10a
D PxVldLst          10a  Const
D PxVldLstLib       10a  Const
D PxUsrId           10a  Const
**-- Change user password: -----**
D ChgUsrPwd      Pr          10i 0
D PxVldLst          10a  Const
D PxVldLstLib       10a  Const
D PxUsrId           10a  Const
D PxUsrPwd          10a  Const
**-- Remove user password: -----**
D RmvUsrPwd      Pr          10i 0
D PxVldLst          10a  Const
D PxVldLstLib       10a  Const
D PxUsrId           10a  Const
**-- Verify user password: -----**
P VfyUsrPwd      B          Export
D              Pi          10i 0
D PxVldL          10a  Const
D PxVldLlib       10a  Const
D PxUsrId          10a  Const
D PxUsrPwd         10a  Const
**-- Verify validation list entry:
D VfyVldLst      Pr          10i 0 ExtProc( 'QsyVerifyValidation+
D                                     LstEntry' )
D VvLstNam        20a  Const
D VvEntId          *  Value
D VvEncDta         *  Value
**
**-- Mainline: -----**
**
C              Eval      EiDta      = PxUsrId
C              Eval      EiDtaLen    = %Len( %TrimR( EiDta ) )
**
C              Eval      EcDta      = PxUsrPwd
C              Eval      EcDtaLen    = %Len( %TrimR( EcDta ) )
**
C              Return     VfyVldLst( PxVldL + PxVldLlib
C                                   : %Addr( EntId )
C                                   : %Addr( EncDta )
C                                   )
**
P VfyUsrPwd      E
**-- Add user password: -----**
P AddUsrPwd      B          Export
D              Pi          10i 0
D PxVldL          10a  Const
D PxVldLlib       10a  Const
D PxUsrId          10a  Const
D PxUsrPwd         10a  Const
D PxUsrDsc         50a  Const
**-- Add validation list entry:
D AddVldLst      Pr          10i 0 ExtProc( 'QsyAddValidation+
D                                     LstEntry' )
D AvLstNam        20a  Const
D AvEntId          *  Value
D AvEncDta         *  Value
D AvEntDta         *  Value
D AvAtrDta        *  Value
**
**-- Mainline: -----**
**
C              Eval      EiDta      = PxUsrId

```

```

C          Eval      EiDtaLen  =  %Len( %TrimR( EiDta ))
**
C          Eval      EcDta     =  PxUsrPwd
C          Eval      EcDtaLen  =  %Len( %TrimR( EcDta ))
**
C          Eval      EdDta     =  PxUsrDsc
C          Eval      EdDtaLen  =  %Len( %TrimR( EdDta ))
**
C          Alloc      15          AdAtrId_p
C          Eval      %Str( AdAtrId_p: 15 ) = 'QsyEncryptData'
**
C          Return     AddVldLst( PxVldL + PxVldLlib
C                               : %Addr( EntId )
C                               : %Addr( EncDta )
C                               : %Addr( EntDta )
C                               : %Addr( AtrInf )
C                               )
**
P AddUsrPwd          E
**-- Get user password: -----**
P GetUsrPwd          B          Export
D          Pi          10a
D PxVldL          10a  Const
D PxVldLlib       10a  Const
D PxUsrId         10a  Const
**-- Find validation list entry:
D FndVldLst        Pr          10i 0 ExtProc( 'QsyFindValidation+
D                               LstEntry' )
D FvLstNam         20a  Const
D FvEntId          *    Value
D FvRtnDta         *    Value
**
**-- Mainline: -----**
**
C          Eval      EiDta     =  PxUsrId
C          Eval      EiDtaLen  =  %Len( %TrimR( EiDta ))
**
C          If          FndVldLst( PxVldL + PxVldLlib
C                               : %Addr( EntId )
C                               : %Addr( RtnDta )
C                               ) = -1
C          Return     *Blanks
**
C          Else
C          Return     %SubSt( RcDta: 1: RcDtaLen )
C          EndIf
**
P GetUsrPwd          E
**-- Change user password: -----**
P ChgUsrPwd          B          Export
D          Pi          10i 0
D PxVldL          10a  Const
D PxVldLlib       10a  Const
D PxUsrId         10a  Const
D PxUsrPwd        10a  Const
**-- Change validation list entry:
D ChgVldLst        Pr          10i 0 ExtProc( 'QsyChangeValidation+
D                               LstEntry' )
D CvLstNam         20a  Const
D CvEntId          *    Value
D CvEncDta         *    Value
D CvEntDta         *    Value
D CvAtrDta         *    Value
**
**-- Mainline: -----**
**
C          Eval      EiDta     =  PxUsrId
C          Eval      EiDtaLen  =  %Len( %TrimR( EiDta ))
**
C          Eval      EcDta     =  PxUsrPwd
C          Eval      EcDtaLen  =  %Len( %TrimR( EcDta ))
**
C          Return     ChgVldLst( PxVldL + PxVldLlib
C                               : %Addr( EntId )
C                               : %Addr( EncDta )

```

```

C                                     : *Null
C                                     : *Null
C                                     )
**
P ChgUsrPwd      E
**-- Remove user password: -----**
P RmvUsrPwd      B      Export
D               Pi      10i 0
D PxVldL         10a  Const
D PxVldLlib      10a  Const
D PxUsrId        10a  Const
**-- Remove validation list entry:
D RmvVldLst      Pr      10i 0 ExtProc( 'QsyRemoveValidation+
D                                     LstEntry' )
D RvLstNam       20a  Const
D RvEntId        *    Value
**
**-- Mainline: -----**
**
C          Eval      EiDta      = PxUsrId
C          Eval      EiDtaLen   = %Len( %TrimR( EiDta ))
**
C          Return    RmvVldLst( PxVldL + PxVldLlib
C                               : %Addr( EntId )
C                               )
**
P RmvUsrPwd      E

```

The above tip was written by Carsten Flensburg. For questions regarding this tip, contact Carsten at <mailto:flensburg@novasol.dk>.

Short Takes:

Source URL: <http://iprodeveloper.com/rpg-programming/apis-example-validation-list-apis>