.NET Admin API

Name Space SecurEnvoy.admin

The SecurEnvoy.admin name space provides basic administration functions to allow user provisioning and ongoing user administration. To use this name space, add a reference to the file adminAPI.dll (located in C:\Program Files (x86)\SecurEnvoy\Security Server\SDK\admin) and declare it, for example with vb.net add the following line: Dim admin As New securenvoy.admin

Note: This API must be located on the same server that has a copy of SecurEnvoy's server software installed.

Note
Setup your project properties to target dot.NET 4.5

Methods

The user object has the following parameters which can be specified (when used in your web page they must be passed with the same names):

Name	Description
adduser()	This method can only be used if the server is setup for SecurEnvoy managed users (Microsoft ADAM). The property sUserID must be set to a unique username. If the property sMobile is not set only basic user information will be created in ADAM. If sMobile is set then the SecurEnvoy attribute will automatically be created. Returns OK, message or ERR,error message.
deluser()	This method can only be used if the server is setup for SecurEnvoy managed users (Microsoft ADAM). The property sUserID must be set to the user that will be deleted. Returns OK, message or ERR,error message.
edituser()	This method is used to change existing user settings. You MUST call listuser first to get the existing user properties before changing them. Returns OK, message or ERR,error message.
getdn(sUserID)	Returns the distinguished name of sUserID or null is the user doesn't exist
Init()	This method is automatically called by other methods to read global configuration settings. Returns OK, message or ERR,error message.
listuser(sdn)	This method is used to set the existing user properties for the user with distinguished name sdn Returns OK, message or ERR,error message.
Randompin()	Returns a cryptographically strong PIN of length set by the server's configuration

Properties

Name	Type	Description
Admin	eNum	ADMIN The user has full admin rights HELPDESK The user has helpdesk only admin rights **Rights** NONE The user has no admin rights UNCHANGED Do not change the existing admin leave
bCreateSeed	Boolean	Soft Token Only, if set to True, creates the first part seed record
bDebug	Boolean	If set to True, debugging is enabled
bFailedReset	Boolean	If set to True resets failed authentication count to 0 after edituser called
bHelpdesk_Enrol	Boolean	If set to True user needs to enrol with secret questions
bHiddenMobile	Boolean	If set to True, the mobile number is encrypted and stored in securenvoys attribute. If set to False, the LDAP attribute "mobile" is used for mobile numbers
bMobile_Enrol	Boolean	If set to True user needs to enrol their mobile phone number
bOffline	Boolean	This applies if Windows Login Agent is deployed or when Windows Integrated Desktop Management is enabled. If set to True this user has a laptop which will require authentication when logging in offline
bResend	Boolean	If set to True, resends the passcode after edituser is called
bResyncToken	Boolean	Resync Soft Token with codes set in sResyncCode1 & sResyncCode2
bSimpleSMS	Boolean	If set to True, SMS messages will not over right or flash. Required for some international SMS gateways to prevent dropping the request
bSoftToken_Enrol	Boolean	If set to True user needs to enrol their soft token app
bTempUser	Boolean	If Set to True User is temporary and will unmanage (or delete if LDAP type is ADAM) after iTempUserDays
bUnmanage	Boolean	If set to True, unmanages this user after edituser is called and releases a license.
bVoiceCall_Enrol	Boolean	If set to True user needs to enrol their phone number
bVoiceCall_HashOnly	Boolean	DO NOT USE RESERVED FOR FUTURE USE
bWeekend	Boolean	Only applies to users in Mode DAYCODE If set to True daycode are sent at weekends (Saturday & Sunday)
Enabled	enum	DISABLED The user is disabled and can't authenticate ENABLED The user is enabled and can authenticate OK ICE The user can only authenticate when ICE is enabled
iNumDays	Integer	Only applies to users in Mode DAYCODE The number of days between each passcode being sent
iTmpDays	Integer	Only applies to users in Mode TMPPW (Temporary Password) The number of days before switching back to One Time Passcode
iTempUserDays	Integer	Only applies to users that have bTempUser set to True The number of days before this user is un-managing or deleted if LDAP type is ADAM
Mode	enum	ONETIME Preload OTP Each passcode can only be used once THREECODES Three Preloaded OTP's in each message REALTIME Passcode Sent in real time after PIN is authenticated SOFTTOKEN Uses soft token apps on smart phones DAYCODE Each passcode can be reused upto iNumDays * 2 VOICECALL VOIP call, user enters the displayed OTP into the phone STATICPW Passcode is static and never changes TMPPW tmp code that switches back to onetime after iTmpDays
sAdministrator	string	The name of the current administrator, used for logging only
sDebugFile	string	The filename to send debug information to (write to c:\debug)
sDN	string	The users distinguished name
sDomainNumber	string	The domain number used when multiple domains are set
sEmail	string	The user's SMTP email address
sFailed	string	The number of failed authentications since the last good one
sFirstName	string	only be used if the server is setup for SecurEnvoy managed users The users first Name
sInstallDir	string	The directory location of the security server (default is to get this setting from the registry)

Name	Type	Description
sLastLogInDate	string	This is set after calling the method listuser or edituser and is a text string of the users last login status
sLastName	string	only be used if the server is setup for SecurEnvoy managed users The users last Name (given name)
sMobile	string	The users mobile phone number
sPassword	string	Only applies to users in Mode STATIC The static passcode to authenticate with
sPin	string	Only applies when PIN management is set to SecurEnvoy The PIN of this user (set to stars of the same length when listing user)
sResyncCode1	string	Soft tokens only, the first 6 digit passcode when bResynToken is True
sResyncCode2	string	Soft tokens only, the next 6 digit passcode when bResynToken is True
sSoftTokenEnrolURL	string	Soft tokens only, this string is set when bCreateSeed in set True
sTmpPassword	string	Only applies to users in Mode TMPPW (Temporary Password) The temporary passcode to authenticate with
sUserID	string	The userID of the current user
SendMode	enum	MOBILE Passcode is sent via SMS to mobile phone EMAIL Passcode is sent via SMTP email

VB .NET Examples

Creating new user fred bloggs when server is configured for internal managed users

```
Dim admin As New securenvoy.admin

admin.bDebug = True ' setup debug, should be set to false on final role-out
admin.sDebugFile = "test_admin.txt" ' this file is created in c:\debug

admin.sDomainNumber = “1”    ' set to your domain number or 1
admin.sUserID = “fbloggs”    ' required userid
admin.sFirstName = “fred”  ' optional first name
admin.sLastName = “blogs”    ' optional last name
admin.sMobile = “447900493049”  ' required mobile phone number
admin.sPin = “4545”   ' optional, defaults to user pin enrolment if not set
admin.Enabled = securenvoy.admin.eEnabled.ENABLED  ' enable this user

Dim sReturn as String = admin.adduser  ' call adduser function

If left(sReturn, 2) = “OK” then
        ‘ adduser returned OK

Else
        ‘ adduser returned an error

End If
```

Delete user fred when server is configured for internal managed users

```
Dim admin As New securenvoy.admin
admin.bDebug = True ' setup debug, should be set to false on final role-out
admin.sDebugFile = "test_admin.txt" ' this file is created in c:\debug

admin.sDomainNumber = “1”    ' set to your domain number or 1
admin.sUserID = “fred” ' userid of user to delete
Dim sReturn as String = admin.deluser

If left(sReturn, 2) = “OK” then
        ‘ deluser returned OK

Else
        ‘ deluser returned an error

End If
```

Enable user fbloggs

```
Dim admin As New securenvoy.admin
admin.bDebug = True 'setup debug, should be set to false on final rollout
admin.sDebugFile = "test_admin.txt" ' this file is created in c:\debug

admin.sDomainNumber = “1”    ' set to your domain number or 1
Dim sdn As String = admin.getdn(“fbloggs”) ' lookup dn from userid
If Left(sdn, 4) = "ERR," Then
  ' getdn returned an error
Else
        admin.listuser(sdn) ' you must call listuser prior to edituser
        admin.Enabled = securenvoy.admin.eEnabled.ENABLED  ' set to enabled

        Dim sReturn as String = admin.edituser() ' call edituser

        If left(sReturn, 2) = “OK” then
          ‘ edituser returned OK

        Else
             ‘ edituser returned an error

        End If
End If
```

Disable user fbloggs

```
Dim admin As New securenvoy.admin
admin.bDebug = True ' setup debug, should be set to false on final rollout
admin.sDebugFile = "test_admin.txt" ' this file is created in c:\debug

admin.sDomainNumber = “1”    ' set to your domain number or 1
Dim sdn As String = admin.getdn(“fbloggs”) ' lookup dn from userid
If Left(sdn, 4) = "ERR," Then
  ' getdn returned an error
Else
        admin.listuser(sdn) ' you must call listuser prior to edituser
        admin.Enabled = securenvoy.admin.eEnabled.DISABLED  ' set to enabled

        Dim sReturn as String = admin.edituser() ' call edituser

        If Left(sReturn, 2) = “OK” then
           ‘ edituser returned OK
        Else
           ‘ edituser returned an error
        End If
End If
```

Create the existing user fbloggs in an external LDAP server such as AD

```
admin.sDomainNumber = “1”    ' set to your domain number or 1
Dim sdn As String = admin.getdn(“fbloggs”) ' lookup dn from userid
If Left(sdn, 4) = "ERR," Then
  ' getdn returned an error
Else
        admin.listuser(sdn) ' you must call listuser prior to edituser
        admin.Enabled = securenvoy.admin.eEnabled.ENABLED  ' set to enabled
        admin.sMobile = “447900493049”  ' required mobile phone number
        Dim sReturn as String = admin.edituser() ' call edituser
        If left(sReturn, 2) = “OK” then
           ‘ returned OK
        Else
           ‘ returned an error
        End If
End If
```

Un-manage the user fbloggs from an external LDAP server such as AD

```
admin.sDomainNumber = “1”    ' set to your domain number or 1
Dim sdn As String = admin.getdn(“fbloggs”) ' lookup dn
If Left(sdn, 4) = "ERR," Then
  ' getdn returned an error
Else
        admin.listuser(sdn) ' call listuser to get existing settings
        admin. bUnmanage = True  ' Unmanage this user
        Dim sReturn as String = admin.edituser() ' call edituser
        If left(sReturn, 2) = “OK” then
     ‘ edituser returned OK
        Else
     ‘ edituser returned an error
        End If
End If
```

Scripting with Microsoft Powershell v3

Note

AdminAPI.dll is a 32bit assembly so you MUST start the 32bit version of PowerShell V3

Start PowerShell V3 - Start – Accessories – Windows PowerShell - Windows PowerShell (x86)
Enter the following commands in PowerShell V3 to load the adminAPI (assumes SecurEnvoy is installed in the default location on a 64bit OS)

```
Add-Type -Path "C:\Program Files (x86)\SecurEnvoy\Security Server\SDK\admin\adminAPI.dll"

$admin = new-object  securenvoy.admin
```

Example: list all methods and properties of AdminDll.dll

```
$admin | Get-Member
```

Example: list the existing user (DN of CN=aaa1,CN=Users,DC=dev,DC=com)

```
$admin.listuser("CN=aaa1,CN=Users,DC=dev,DC=com")

 $admin
```

Example: change an existing user (DN of CN=aaa1,CN=Users,DC=dev,DC=com) mobile number to 123456

```
$admin.listuser("CN=aaa1,CN=Users,DC=dev,DC=com")

$admin.sMobile="123456"

$admin.edituser()
```

Example: list the existing user with a UserID of aaa1 (Note required version 7.2 or higher)

```
$admin.listuser($admin.getdn("aaa1"))

$admin
```

Example: change an existing user (DN of CN=aaa1,CN=Users,DC=dev,DC=com) Admin to FULL

```
$admin.listuser("CN=aaa1,CN=Users,DC=dev,DC=com")

$admin.Admin = ([securenvoy.admin+eAdmin]::FULL)

$admin.edituser()
```

Example: change an existing UserID aaa1 to Disabled (Note required version 7.2 or higher as getdn is used)

```
$admin.listuser($admin.getdn("aaa1"))

$admin.Enabled = ([securenvoy.admin+eEnabled]::DISABLED)

$admin.edituser()
```