NTSecurity Class

The NT security class provides a way to programmatically manipulate user accounts and global groups in a Windows NT domain.   It consists of the following three related objects:

- NTDomain
- NTGroup
- NTUser

NTDomain Object

The **NTDomain** object provides information about a domain, including users and groups.

Properties

Name	Type	Read	Write	Description
Groups	NTGroup	✓		Represents all the global groups in the domain specified by sDomain.
eLogging	eLogLevel	✓	✓	Returns or sets the logging level for the object. Can be: 0 – *LogNone* No logging 1 – *LogAddDel* Logs user and group adds/deletes 2 – *LogFull* Logs all changes to users/groups All entries are written to the Application log on the local system. *LogAddDel* is the default setting.
sDC	String	✓		The domain controller for the domain specified by sDomain.
sDomain	String	✓	✓	Returns or sets the name of the domain to manipulate. Note that you cannot change the name of a domain; setting this property will select a different domain.
sLastError	String	✓		The results of the last action. Will hold "SUCCESS" or a description of the error that occurred.
Users	NTUser	✓		Represents all the user accounts in the domain specified by sDomain.

Methods

None.

Events

None.

Usage

The following shows how to use the NTDomain object.

```
Sub GetDomainInfo()
     Dim MyDomain as New NTDomain, MyGroups() as String, i as Integer

     ‘ Get some information about the “TEST” domain.
     MyDomain.sDomain = “TEST”
     Debug.Print “The TEST domain’s PDC is “ & MyDomain.sDC

     ‘ Get a list of groups in the domain
     MyDomain.Groups.Enumerate MyGroups
     For i = 1 To UBound(MyGroups)
          Debug.Print MyGroups(i)
     Next i

End Sub
```

NTGroup Object

The **NTGroup** object represents the global groups in a domain.

Properties

Name	Type	Read	Write	Description
oParent	NTDomain	✓		A reference to the NTDomain object that owns this instance.
sComment	String	✓	✓	Returns or sets the description for the group specified in sGroupName.
sGroupName	String	✓	✓*	Returns or sets the name of the domain to manipulate. *Note that you cannot change the name of a group; setting this property will select a different group.
sLastError	String	✓		The results of the last action. Will hold "SUCCESS" or a description of the error that occurred.
Users	NTUser	✓		Represents all the user accounts in the group specified by sGroupName.

Methods

Name	Parameters	Returns	Description
Add	sGroupName As String, sComment As String	Boolean	Adds a global group to the domain.
Delete	sGroupName as String	Boolean	Removes a global group from the domain.
Enumerate	sList() as String	-	Lists all groups in the domain.

Events

None.

Usage

All properties will be NULL until you select a group by setting the value of the sGroupName property. Only global groups are accessible.

The following shows how to use the NTGroup object.

```
Sub PlayWithGroup()
     Dim MyDomain as New NTDomain

     ‘ Get some information about the “TEST” domain.
     MyDomain.sDomain = “TEST”

     ‘ Create “TestUsers” global group and then select it.
     MyDomain.Groups.Add(“TestUsers”, “Test User Group”)
     MyDomain.Groups.sGroupName = “TestUsers”

     ‘ Did it work?
     if MyDomain.Groups.sLastError = “SUCCESS” then
          ‘ Add user “SmithJ” to group.
          MyDomain.Groups.Users.Add “SmithJ”
          ‘ Lastly, change the comment.
          MyDomain.Groups.sComment = “Test User Group with SmithJ”
     Else
          MsgBox “Unable to add group – “ & MyDomain.Groups.sLastError
     End If

End Sub
```

NTUser Object

The **NTUser** object represents the users in its parent (either an NTDomain or NTGroup).

Properties

Name	Type	Read	Write	Description
bDisabled	Boolean	✓	✓	Retuns or sets whether the account is disabled.
bLockedOut	Boolean	✓		Returns whether the account is locked out.
ePrivilege	ePriv	✓		Returns the privilege level of the user specified in sUserName. Can be: 0 – *privGuest* Guest level account 1 – *privUser* User level account 2 – *privAdmin* Administrator level account
oParent	Object	✓		A reference to the parent, either an NTDomain or NTGroup object, that owns this instance.
sComment	String	✓	✓	Returns or sets the description of the user specified in sUserName.
sFullName	String	✓	✓	Returns or sets the full name (i.e. “John Smith”) of the user specified in sUserName.
sHomeDir	String	✓	✓	Returns or sets the home directory of the user specified in sUserName.
sLastError	String	✓		The results of the last action. Will hold "SUCCESS" or a description of the error that occurred.
sPassword	String		✓	Sets the password for the user specified in sUserName.
sScriptPath	String	✓	✓	Returns or sets the logon script for the user specified in sUserName.
sSID	String	✓		Returns a string representation of the SID for the user specified in sUserName.
sUserName	String	✓	✓*	Returns or sets the name (user ID) of the user (i.e. “SmithJ”) to manipulate. *Note that you cannot change the name of a user; setting this property will select a different user.

Methods

Name	Parameters	Returns	Description
Add	sUserName As String, *Optional sPWD As String,* *Optional sHomeDir As String,* *Optional sComment As String,* *Optional sScript As String,* *Optional sFName As String,* *Optional ePrivLevel As ePriv,* *Optional bDis As Boolean*	Boolean	Adds a user to the parent (either an NTDomain or NTGroup object). The optional parameters must be specified if adding a user to a domain.
Delete	sUserName as String	Boolean	Removes a user from the parent (either an NTDomain or NTGroup object).
Enumerate	sList() as String	-	Lists all users in the parent (either an NTDomain or NTGroup object).
UnlockAccount		-	Unlocks the account specified in sUserName.

Events

None.

Usage

All properties will be NULL until you select a user by setting the value of the sUserName property.

The following shows how to use the NTUser object.

```
Sub PlayWithUsers()
     dim MyDomain as New NTDomain, MyUsers() as String, i as Integer

     ‘ Get some information about the “TEST” domain.
     MyDomain.sDomain = “TEST”

     ‘ Create “TestUser” user account and then select it.
     MyDomain.Users.Add(“TestUser”, “firstpassword”, “\\SERVER\TESTUSER”, _
                        “Test User Account”, “LOGIN.BAT”, “Test A. Count”, _
                        privUser, False)
     MyDomain.Users.sUserName = “TestUser”

     ‘ Did it work?
     if MyDomain.Users.sLastError = “SUCCESS” then
          ‘ Get the SID for the user.
          MsgBox “The SID for “ & MyDomain.Users.sUserName & “ is “ & MyDomain.Users.sSID
          ‘ Now reset his password.
          MyDomain.Users.sPassword = “secondpassword”
     Else
          MsgBox “Unable to add user – “ & MyDomain.Users.sLastError
     End If

     ‘ Select the group “BadBoys”...
     MyDomain.Groups.sGroupName = “BadBoys”
     ‘ ...and get all the users...
     MyDomain.Groups.Users.Enumerate MyUsers
     ‘ ...and then disable them!
     For i = 1 To UBound(MyUsers)
          MyDomain.Groups.Users.bDisabled = True
     Next i

End Sub
```